

**ВАЗОРАТИ МАОРИФ ВА ИЛМИ ҶУМҲУРИИ ТОҶИКИСТОН
ДОНИШКАДАИ МОЛИЯ ВА ИҚТИСОДИ ТОҶИКИСТОН**

Муллошараф Арабов

**АСОСҲОИ БАҶНОМАСОЗИ ДАР
ЗАБОНИ C++**

ББК 32.973+74.202.4+92Я2

А-73

ISBN 978-99975-850-7-3

Аз тарафи Шӯрои илмӣ–методи Донишкадаи молия ва иқтисоди Тоҷикистон санаи «26» сентябр соли 2017 таҳти рақами 3/5.4 ба чоп тавсия шудааст.

Муллошараф Арабов. Асосҳои барномасозӣ дар забони C++. Душанбе. РТСУ 2018. - 421с.

Дар ин китоб асосҳои барномасозӣ дар забони C++ ба таври комил оварда шудааст. Бинобар ин, хонандагон метавонанд, бе ягон душворӣ ва доштани маълумоти иловагӣ, ба ғайр аз информатикаи мактабӣ, асосҳои барноманависиро ба таври комил омӯзанд.

Гарчанде ки ин китоб барои донишҷӯёни мактабҳои олӣ навишта шудааст, аз метавонанд хонандагони синфҳои болоии мактабҳои миёна, миёнаи махсус, литсейву коллеҷҳо ва умуман касонеки хоҳиши омӯختани барноманависиро доран, низ ситифода баранд.

Муқарризон: **Ашӯров Хуршед Мирумарович.**, номзади илмҳои иқтисодӣ, мудири кафедраи технологияи иттилоотӣ ва иртиботии ДМТ

Рачабов Фахриддин Сайфиддинович, номзади илмҳои физика ва математика, мудири кафедраи моделсозии математикӣ ва иттилоотии ДМИТ

Мухаррирон: **Саидолимов Комилҷон**, номзади илмҳои филология, дотсент

Халилова Моҳчеҳра Шавкатовна, номзади илмҳои физика ва математика, дотсент

Мундарича

Пешгуфтор	12
Боби 1. Маълумоти умумӣ дар бораи забони барномасозии C++	14
Таърихи пайдоиши забони C++	14
Алифбои забони C++	16
Ҳассосият ба ҳуруфот ва эзоҳ дар забони C++	16
Калимаҳои калидии забони C++	16
Идентификаторҳо (identifiers)	17
Имкониятҳои нави C++	17
Компилятор ва интерпритатор	18
Муҳити интегронидашудаи коркард (МИК)	21
Равзанаи таҳриргари коди лоиҳа ё сохтори барнома	21
Китобхонаҳои (Библиотекаҳои) C++	22
Саволҳо барои мустаҳкамкунӣ	23
Боби 2. Додаҳо, тағйирёбандаҳо ва амалҳо	24
Муқаддима	24
Додаҳои навъи (типи) рамзӣ	25
Додаҳои навъи бутун	25
Додаҳои навъи ҳақиқӣ	26
Додаҳои навъи мантиқӣ	27
Додаҳои навъи ҳолӣ	27
Тағйирёбандаҳо ва муаррифӣ намудани онҳо	27
(variables declaration)	27
Қимати аввала бахшидан ба тағйирёбандаҳо	30
Аз як навъ ба навъи дигар табдил додани тағйирёбандаҳо	30
Амалҳо	31
Амалҳои арифметикӣ (arithmetic operators)	31
Амалҳои мантиқӣ (logical operators)	35
Амали шартӣ (conditional operator) ё сегона (тернерӣ)	36
Амали коммо (The comma operator) ё вергул	37
Амалҳои ягона (Unary operators)	37
Амалҳои ҳофиза	38
Муродифи амалҳо	40
Ифодаҳо (expression)	41
Операторҳо	42
Функсияҳои стандартӣ дар забони C++	42
Саволҳо барои мустаҳкамкунӣ	43
Боби 3. Воситаҳои дохилкуниву хориҷкунӣ дар C++	45
Муқаддима	45
Функсияҳои дохилкуниву хориҷкунии қолабдор (formatted console input/output) ..	45

Функсияи printf()	46
Функсияи scanf().....	49
Функсияи getchar() ва putchar().....	49
Функсияҳои дохилкунӣ ва хориҷкунӣ дар забони C++	50
Функсия (Объект)-ҳои хориҷкунӣ.....	50
Функсияи put.....	52
Функсияҳои get ва getline	52
Беқолаббанди дохил ва хориҷ намудани маълумотҳо	54
Ҳарфҳои кириллӣ дар C++	55
Саволҳо барои мустақамкунӣ	57
Намунаи ҳалли масъалаҳо	58
Масъалаҳо барои кори мустақилона	61
Масъалаҳои гурӯҳи Begin	61
Масъалаҳои гурӯҳи Integer.....	62
Масъалаҳои гурӯҳи Boolean.....	64
Боби 4. Сохтори операторҳои идоракунии	66
Муқаддима	66
Операторҳои шартӣ.....	66
Операторҳои шартии мураккаб(nested if-else)	68
Оператори switch	70
Операторҳои такрорӣ.....	73
Оператори даврӣ бо шартӣ пешоянд	73
Оператори даврии бо шартӣ пасоянд do/while	74
Оператори даврии параметрдор	75
Сохтори даврҳои мураккаб ё дохилиҳам.....	77
Оператори коммо (the comma operators).....	79
Операторҳои continue ва break	79
Оператори гузариши бешарт (goto).....	80
Саволҳо барои мустақамкунӣ	81
Намунаи ҳалли масъалаҳо	82
Масъалаҳо барои кори мустақилона	87
Масъалаҳои гурӯҳи If	87
Масъалаҳои гурӯҳи Case	88
Масъалаҳои гурӯҳи For.....	90
Масъалаҳои гурӯҳи While	92
Боби 5. Функсияҳо(functions).....	94
Муқаддима	94
Тарзи муаррифӣ (declaration) намудани функсияҳо.....	97

Оператори return	98
Ба функсия равон кардани параметрҳо	100
Функсияҳои бозгаштӣ (Рекурсивний функция)	102
Пешнамунаҳои аз функсия (Function Prototypes)	103
Истифодаи якчанд функсия дар як барнома	104
Қаламрави тағйирёбандаҳо	105
Class-ҳои ҳофиза	107
Синфи ҳофизаи худкор (automatic)	107
Синфи ҳофизаи доимӣ (static)	108
Синфи ҳофизаи register (register)	109
Ба аргументҳои функсия бахшидани қимати аввала	110
Сарборгузори функсияҳо	111
Қолаби функсияҳо	112
Саволҳо барои мустақамкунӣ	114
Намунаи ҳалли масъалаҳо	114
Масъалаҳо барои кори мустақилона	122
Боби 6. Массивҳо (arrays)	124
Муқаддима	124
Эълон ва ё муаррифӣ намудани массивҳо	124
(declaration arrays)	124
Ба массивҳо бахшидани қимати аввала	125
(initialization arrays)	125
Доҳилкунӣ ва хоричкунии элементҳои массив	126
Сумма ва ҳосили зарби элементҳои массив	127
Ба намуди диаграмма тасвир намудани элементҳои массив	128
Бо ададҳои тасодуфи пур намудани элементҳои массив	129
Чустучӯи элементҳои калонтарин ва рақами тартибии он дар массив	130
Равон кардани массив ба функсия ба сифати аргумент	131
Массивҳои динамикӣ	132
Илова ва нест намудани элемент дар массив	134
Массивҳо аз сатрҳои матнӣ	136
Мураттабсозӣ ва чустучӯии элементҳо дар массив	138
Гурӯҳбандии равишҳои мураттабкунӣ	138
Мураттабсозии ҳубобӣ (bubble sort)	139
Мураттабсозӣ бо равиши интиҳоб (selection sort)	141
Мураттабсозии гузориши (insertion sort)	142
Мураттабсозии tag (tag sort)	143
Мураттабсозии босуръат ё зуд (quick sort)	143
Чустучӯ кардан (searching)	145
Чустучӯи хаттӣ	146

Чустучӯи равиши дуй (binary search).....	146
Чустучӯи равиши тасодуфӣ	148
Массивҳои бисёрченака.....	148
Ба массивҳои дученака бахшидани қимати аввала	149
Истифодаи массивҳои дученака	150
Матритсаҳои рамзӣ	150
Ғалбери Эратосфен.....	151
Ғалбери Сундарама	152
Намунаи ҳалли масъалаҳо	153
Масъалаҳо барои кори мустақилона	162
Масъалаҳои гурӯҳи Array.....	162
Масъалаҳои гурӯҳи Matrix	163
Боби 7. Ишорагарҳо (Pointers)	165
Муқаддима	165
Муаррифӣ намудани тағйирёбандаҳои навъи ишорагар.....	166
Ёфтани ё пайдо кардани суроғи як тағйирёбанда.....	167
Ба ишорагар бахшидани қимати аввала.....	169
Даъвати функсия аз рӯи суроға ва баргардонидани қимат	170
Ишорагар бар ишорагар (Pointer to Pointer)	172
Истифодаи ишорагарҳо барои кор кардан бо массивҳои дученака	173
Истифодаи ду ишорагар барои массивҳои дученакаи динамикӣ	175
Массивҳо ва ишорагарҳо	177
Равон кардани массив ба функсия ба сифати аргумент.....	179
Аз функсия баргардонидани суроға.....	182
Таҷҳизоти хотираи ғаврӣ.....	183
Саволҳо барои мустаҳкамкунӣ	185
Масъалаҳо барои кори мустақилона	186
Масъалаҳои гурӯҳи ишорагар ва суроға.....	186
Масъалаҳои гурӯҳи массиви динамикӣ	190
Масъалаҳои гурӯҳи ишорагарҳо ва мурағабсоӣ	191
Боби 8. Кор бо сатрҳо ва рамзҳо.....	194
Муқаддима	194
Ба тағйирёбандаҳои рамзӣ бахшидани қимати аввала	194
Функсияҳои кор бо рамзҳо.....	196
Функсияи get ва getline	198
Функсияҳои табдилдиҳии сатрҳо	199
Истифодаи функсияҳои табдилдиҳии сатрҳо	200
Функсияҳои чустучӯи аз китобхонаи кор бо сатрҳо.....	201
Функсияҳои кор бо ҳофиза аз китобхонаи коркарди сатрҳо	203
Функсияҳои кор бо сатрҳо ва рамзҳо.....	205

Истифодаи функсияи strcpy() ва strcat ()	207
Истифодаи функсияи strset(char_c, int_ch).....	207
Массивҳои дученака аз навъи рамзӣ.....	208
Равон кардани сатрҳо ба функсия ба сифати аргумент	208
Синфи нави кор бо сатрҳо дар забони C++	209
Сохтани сатр	209
Кор бо тағйирёбандаҳои навъи string.....	210
Доҳилкунӣ ва хоричкуни додаҳои навъи string	210
Дастрасӣ ба символҳои тағйирёбандаҳои навъи string	212
Амалиёт ба сатрҳо	213
Функсия-аъзоҳои кор бо сатрҳо.....	213
Функсияи assign()	214
Функсияи append().....	215
Функсияҳои insert(), erase() ва replace()	216
Функсияи at()	218
Мисоли функсияи find_last_of()	218
Истифодаи функсияи find() ва copy().....	219
Функсияҳои empty() ва erase(char *s).....	220
Функсияи substr(pos,n)	221
Функсияи push_back.....	221
Функсияи swap.....	222
Функсияи c_str	222
Функсияи rbegin.....	223
Функсияи clear	223
Функсияи pop_back	224
Саволҳо барои мустаҳкамкунӣ	224
Намунаи ҳалли масъалаҳо	225
Масъалаҳо барои кори мустақилона	229
Боби 9. Сохторҳо (structures).....	231
Муқаддима	231
Муаррификунии навъи сохтор.....	231
Дастраси ба майдонҳои сохтор	232
Ба ҳамдигар бахшидани сохторҳо	233
Доҳил ва хорич намудани тағйирёбандаҳои сохтор	233
Ба аъзоҳои сохтор бахшидани қимати аввала.....	234
Методҳо. Муаррифӣ намудани методҳо.....	235
Конструктор. Истифодаи конструктор	236
Массивҳо аз навъи сохтор	237
Таҳлили як сохтор	239
Ишорагар ба сохтор (pointer to struct)	242

Равон кардани аъзоҳои сохтор ба функсия	243
Ишорагари сохторро баргардонидани функсия	245
Ба сифати аргументи функсия равон кардани сохтор.....	246
Навъи додаҳое, ки ба воситаи корбар муаррифи мегарданд (user-defined data types)	250
Сохтори додаҳо ва ишорагарҳо	251
Ишорагарҳо ба сифати элементи сохтор	252
Иттиҳодҳо (Unions)	253
Додаҳои навъи шуморишӣ	256
Саволҳо барои мустаҳкамкунӣ	257
Мисолҳои барои кори мустақилона	258
Боби 10. Файлҳо.....	260
Муқаддима	260
Файлҳо ва ҷараёнҳо (Files and streams).....	262
Ҷараёнҳои стандартӣ (standard streams).....	262
Ҷараёни матнӣ (text stream).....	263
Ҷараёнҳои бинарӣ ё дӯй	263
Фарқияти асосӣ байни додаҳои матнӣ ва бинарӣ ё дӯй	263
Кор бо файлҳои матнӣ	264
Сохтани як файли матнӣ ва навиштани додаҳо дар он.....	264
Хондани додаҳо аз як файли матнӣ.....	266
Кор бо файлҳои бинарӣ ё дуӣ	269
Системаи I/O-и буфердор (buffering I/O system)	269
Ишорагари файл (File pointer).....	270
Кушодан ва пӯшидани як файли бинарӣ.....	270
Функсияи putc()	273
Функсияи getc().....	273
Функсияи feof().....	274
Функсияи freopen().....	275
Истифодаи массивҳои сеченака.....	275
Функсияи fclose().....	276
Функсияи ferrror()	277
Функсияҳои fscanf() ва fprintf().....	277
Функсияҳои remove() ва rename().....	279
Функсияи fseek().....	281
Саволҳо барои мустаҳкамкунӣ	282
Намунаи ҳалли масъалаҳо	283
Масъалаҳо барои кори мустақилона	293
Боби 11. Барноманависии динамикӣ.....	295
Муқаддима	295

Листҳои пайвандӣ	295
Эълон намудани Node	296
Пайваст намудани гиреҳои листҳои якпайвандӣ	297
Эҷоди листҳои якпайвандӣ	298
Нест намудани гире аз листи пайвандӣ	303
Иҷрои амалиётҳои дигар бо листҳои якпайвандии	305
Листҳои дупайвандӣ	308
Пуштаҳо (Stack)	311
Пиедасозии пушта бо массив	312
Пиедасозии пушта бо листҳои пайвандӣ	313
Сафҳо	315
Саволҳо барои мустаҳкамкунӣ	316
Мисолҳо барои кори мустақилона	316
Боби 12. Китобхонаи қолаб (шаблон)-ҳои стандартӣ STL (Standard Template Library)	318
Муқаддима	318
Назардошти умуми дар бораи библиотекаи STL	319
Объектҳои функционалӣ	320
Объектҳои функционалии арифметикӣ	321
Предикатҳо	321
Инкоркунандаҳо	322
Пайвасткунандаҳо	323
Адаптерҳои ишорагар ба функсия	324
Адаптери методҳо	325
Контейнерҳои class-ҳо	326
Векторҳо	327
Массивҳо ба сифати элемент дар вектор	330
Дастрасшавӣ ба элементҳои вектор тавассути итераторҳо	331
Функсияҳои insert() ва erase ()	332
Рӯйхатҳо	333
Мураттабкунии рӯйхатҳо	338
List-ҳо аз массив	339
Якҷоя кардани ду List	339
Тасвирҳо (Таҷассумҳо)	341
Алгоритмҳо	346
Истифодаи алгоритмҳо дар амалия	351
Алгоритми count ва count_if()	352
Алгоритми find ва find_if	354
Алгоритми adjacent_find	354
Алгоритми search ва search_n	356

Алгоритми merge	356
Алгоритми next_permutation().....	357
Алгоритми sort.....	358
Алгоритми swap().....	359
Алгоритми remove_copy() ва алгоритми replace_copy().....	359
Алгоритми transform()	361
Алгоритми random_shuffle().....	362
Алгоритми accumulate ().....	363
Алгоритмҳои copy() ва copy_backward()	363
Алгоритми replace, replace_if, replace_copy ва replace_copy_if	366
Алгоритми generate ва generate_n.....	367
Алгоритми fill ва fill_n.....	367
Алгоритми partial_sort ва partial_sort_copy	368
Алгоритми nth_element.....	369
Алгоритмҳои кор бо маҷмӯъҳо.....	370
Алгоритми includes	371
Алгоритми set_insersection.....	372
Алгоритми set_defference, set_symmetric_difference	372
Алгоритми set_union	374
Саволҳо барои мустаҳкамкунӣ	375
Намунаи ҳалли мисолҳо	376
Дохилкунӣ	378
Хориҷкунӣ.....	378
Матри барнома	378
Мисолҳо барои кори мустақилона	378
Боби 13. Графҳо	379
Муқаддима	379
Дараҷаи қуллаҳои графҳо.....	380
Шакли геометрии графҳо	380
Намоиши дискретии графҳо.....	385
Матритсаи ҳамсоғӣ.....	385
Матритсаи якзайлӣ.....	386
Матритсаи вазнӣ.....	387
Рӯйхати теғҳо	388

Чустучӯ дар умк (Depth-first search).....	388
Татбиқи алгоритми чустучӯ дар умк.....	390
Чустучӯ дар васеъгӣ.....	394
Сохтони роҳи кутоҳтарин.....	397
Намунаи ҳалли мисолҳо	403
Дарахтҳо	404
Дарахтҳои дӯӣ.....	405
Сохтори гиреҳи дарахти дӯӣ.....	406
Чустучӯ (паёмиш)-и дарахти дӯӣ	409
Паёмиши inorder	409
Паёмиши postorder.....	410
Паёмиши postorder.....	410
Саволҳо барои мустаҳкамкунӣ	414
Мисолҳо барои кори мустақилона	414
Манбаъҳои истифодашуда	417

Пешгуфтор

Ҳамқадам бо асру бо даврон мебояд будан,
Бо хабар аз илму аз ирфон мебояд шудан.
Ҳеч роҳе нест, ки бепоён бошад дар ҷаҳон,
Бо ирода бо ҳавас бо ҷон мебояд шудан.

(Муллошараф Арабов)

Барномасозӣ низ ҳунар буда, барои бартараф намудани баъзе мушкилоти ҷомеа бо истифода аз ягон забони барноманависӣ барнома навишта мешавад. Ҳангоме, ки шахс барнома менависад, метавонад ба компютер ҳамчун раис (роҳбар) муомила кунад. Барои навиштани барнома забонҳои барноманависии зиёде мавҷуданд, ки аз байни онҳо якчанд забон маъмулу машҳур ва пурқудрат буда, мавриди писанди барноманависон ва омӯзандагон мебошад. Забони C++ аз зумраи чунин забонҳост. Забонҳои барноманависӣ бо гузашти замон такмили густариш меёбанд. Забони C++ низ дар айни ҳол дар ҳолати рушд ва густариш мебошад.

Забони C++ дорои қудрати зиёд буда, метавонад ҳам дар муҳити Windows ва ҳам дар муҳити дигар системаҳои омилӣ кор кунад.

Забони C++ ҳамаи имкониятҳои забони C-ро доро буда, илова бар ин, қудрати БОН-ро низ соҳиб аст. Баъзеҳо то кунун ҳарчанд ки аз забони C истифода баранд ҳам, вале забони C++ дар айни ҳол густариш меёбад.

Пас аз соҳибистиклол гардидани Тоҷикистон чи дар мактабҳои миёна ва чи дар коллеҷу донишгоҳҳо омӯзиши компютер ва барноманависӣ ба роҳ монда шудааст, вале мутаасифона, то кунун дар Тоҷикистон адабиёти фароғири ин соҳа бо забони давлатӣ (модарӣ) вучуд надорад. Муаллиф кӯшиш ба харҷ додааст, ки бо забони модарӣ барои хонандагон ҳамчунин дастуреро пешкаш намояд.

Ин китоб тарзе навишта шудааст, ки хонандагони он метавонанд бе ягон душворӣ ва доштани маълумоти иловагӣ ба ғайр аз информатикаи мактабӣ, асосҳои барноманависиро ба таври комил омӯзанд. Барои ин зарур аст, ки ҳар мавзӯро бодикқат хонда, барномаҳои овардашударо таҳлил кунанд. Дар охири ҳар боб барои мустақамкунӣ саволҳо ва мисолҳои мувофиқ барои намуна ва барои кори мустақилона оварда шудааст. Мақсади овардани ин мисолҳо аз он иборат аст, ки хонанда дониши омӯхташро санчида, маҳорати барноманависиаширо сайқал диҳад.

Китоби мазкур 13 бобро дарбар гирифта, ҳамчун воситаи муҳиме барои худомӯзии асосҳои барноманависӣ дар C++ ба ҳисоб меравад. Аз ин китоб хонандагон метавонанд, дар як муддати кӯтоҳ асосҳои забони C++-ро ба таври комил омӯхта, заминае барои омӯзиши забонҳои C# ва Java пайдо кунанд.

Бояд қайд кард, ки ин китоб на танҳо ҳамчун воситаи худомӯзӣ, балки воситаи методиву таълимӣ низ мебошад. Ҳамаи барномаҳои ин китоб дар компютер иҷро ва тест карда шудаанд.

Аз сабаби он ки дар забони C++ истифодабарии имкониятҳои забони C вучуд доранд, бинобар ин, баъзе хусусиятҳои забони C низ оварда шудаанд.

Дар охири китоб феҳристи васеи адабиёт чой дода шудааст, ки барои пурратар ва ҳаматарафа омӯхтани забони C++ кўмак мекунад.

Инак, дастур пешкаши хонандагон мегардад, ки он нахустин таълифоти муаллиф буда, аз нуқсону ғалатҳо орӣ нест. Аз ин рӯ, муаллиф аз ҳамаи хонандагон эҳтиромона хоҳиш менамояд, ки ҳар гуна фикру мулоҳизоти хешро оид ба ин дастур ба нишонаи электронии *cool.araby@mail.ru* ирсол намоянд.

Мо умедворем, ки фикру дархости беғарзонаи Шумо барои беҳтар шудани сифати китоб дар оянда кумак хоҳанд кард.

Муаллиф ба шахсоне, ки фикру мулоҳизаҳои худро оид ба мазмуни китоб беғарзона ба ӯ мерасонанд, қаблан миннатдории худро баён месозад.

Муаллиф ба муқарризон Ашӯров Х.М., Раҷабов Ф.С., муҳаррирон Саидолимов К.С., Халилова М.Ш. ва шахсоне, ки дар рафти навиштани китоб маслиҳатҳои муфид дода, тавачҷуҳ зоҳир намуданд, миннатдории худро баён месозад.

Боби 1. Маълумоти умумӣ дар бораи забони барномасозии C++

Таърихи пайдоиши забони C++

Забони C++ -ро соли 1979 дар фирмаи «Bell Laboratories» (шаҳри Мурей-Хилли иёлоты Нью-Черси) Берн Страуструп сохта, онро C бо классҳо (C with classes) номид. Мақсади чунин ном гузоштани забон аз он иборат буд, ки таҳкурси ва заминаи пайдоиши он забони C ба ҳисоб мерафт. Забони C-ро бошад, соли 1976 Данис Ричард тарроҳӣ кардааст. Забони мазкур дар асоси забони BSPL, ки онро Томпсон тарроҳӣ карда буд, сохта шудааст. Сабаби асосии C ном гирифтани забон дар он буд, ки он пас аз забони B пайдо шудааст.

Идеяи сохтани забони нав ба Страуструп дар вақти навиштани диссертатсияаш оғоз ёфт. Вай муайян кард, ки забони моделсозии Si-mula имкониятҳои зиёдеро доро асту, вале сушт кор мекунад. Аз ин рӯ, дар забони C имкониятҳои онро илова намуда, онро C бо классҳо (C with classes) номид.

Забони C БОН-ро пуштибонӣ намекард. Страуструп забони C++ ро тавре сохт, ки қариб пурра БОН мебошад. Имконияти БОН-и забони C++ ро Страуструп аз дигар забонҳо ба хусус забони Simula 67 гирифт.

Дар забони C бо классҳо имкониятҳои нав ба монанди функцияҳои виртуалӣ, сарборӣ ниҳодан ба функцияҳо ва операторҳо, услуби нави эзоҳи («//») ва ғайра илова гардиданд. Баъд аз ин иловакуниҳо соли 1983 номи он ба C++ иваз карда шуд.

Моҳи октябри соли 1958 нахустин версияи забони C++ ба фурӯш бароварда шуд. Аввалин китоби C++ бошад, бо номи «Забони бар-номанависии C++» низ дар ҳамин сол аз чоп бароварда шуд. Соли 1982 версияи дуҷуми забони C++ бароварда шуд.

Соли 1998 Комитети ANSI/ISO забони C++ -ро стандарти байналхалқӣ (ISO/IEC 14882:1998 «Standard for the C++ Programming Language») қунонд. Баъд аз анҷоми кор дар болои лоиҳаи стандартқунонии C++ ҳодисаҳои наве ба амал омада, стандартӣ C++-ро маҷбур ба васеъшавӣ намуданд. Масалан, Александр Степанов китобхонаи стандартии қолибҳо (Standard Template Library-STL)-ро сохт. Пас аз сохтани китобхонаи мазкур, Комитети ANSI/ISO барои STL-ро ба C++ илова намудан розӣ шуд.

Комитети ANSI/ISO аз болои C++ чанде кор карда, ҳатогиҳои C++-ро ислоҳ намуд ва дар натиҷа, соли 2003 версияи ислоҳшудаи стандарти C++ (ISO/IEC 14882:2003) бароварда шуд. Стандарти мазкур аз ду қисми асосӣ иборат буд: эзоҳи ядроӣ (ҳастаи) забон ва эзоҳи китобхонаҳои стандартӣ.

Соли 2005 китобхонаи техникаи ҳисобот 1 (Library Technical Report 1 (TR1)) пайдо шуд.

Соли 2009 кӯшиши иваз (аз нав) кардани стандарти пешинаи забон ба миён омад. Дар аввал стандарти C++99 мавҷуд буд, баъд стандарти C++11 пайдо шуд. Имрӯзҳо бошад, аз стандарти - C++11 истифода мебаранд, ки он яке аз стандартҳои васеъшуда ба ҳисоб рафта, дар дохили он китобхонаҳои стандартӣ аз он ҷумла TR1 низ мавҷуд аст.

Мувофиқи стандарти ISO/ IEC 14882, забони C++ ба ду қисм— «Барноманависии сохторӣ(Структурное программирование)» ва «Барноманависии ба объектҳо нигаронидашуда (Объектно-ориентированное программирование)» ҷудо карда шудааст.

Дар ин дастур имкониятҳои қисмати «Барноманависии сохторӣ» мавриди баҳс қарор дода мешавад.

Ин китоб қариб, ки фарогири ҳамаи стандартҳо буда, барномаҳои дар он овардашуда метавонанд дар компиляторҳои зиёди забони C++ ба монанди: Borland C++ Builder, Microsoft C++, Bloodshed Dev C++, Code Block C++, Turbo C++, CodeGear C++, CodeGear XE C++ ва ғайра кор кунанд.

Баъзе хусусиятҳои забони C++

Забони C++ дорои хусусиятҳои зиёде мебошад, ки баъзе аз онҳо зимнан оварда шудаанд:

C++ забони маҳбубу писандида буда, ниҳоят қудратманд мебошад. Дар он ҳеч маҳдудият барои барноманависӣ вуҷуд надорад. Аз ин рӯ, ҳар он ки дар зехни барномасоз пайдо мешавад, онро метавонад дар C++ амалӣ гардонад.

C++ бо забони Асамблер робитаи наздик дорад. Аз ин рӯ, дар он метавон имкониятҳои забони Асамблерро низ истифода кард.

C++ забони ба объектҳо нигаронидашуда мебошад.

C++ нисбат ба регистр ҳассос мебошад. Дар забони мазкур навишти ҳарфҳои хурд аз калон фарқ доранд. Масалан: fog як кали-маи калидӣ аст, вале FOR не. Аз ин рӯ, ба хонандагон тавсия карда мешавад, ки ҳангоми барноманависӣ аз ҳарфҳои хурд истифода кунанд.

C++ забони барноманависии система аст. Барномаҳои система барномаҳои ҳастанд, ки онҳоро дар дискҳо гирифта, ба компютер насб мекунанд. Ба ин мисол шуда метавонанд:

1. Системаҳои омилӣ
2. Компиляторҳо ва интерпретаторҳо
3. Вожапардозҳо
4. Мудирияти бонкҳои иттилоотӣ ва ғайра.

C++ дорои хусусияти хосе мебошад. Ҳар барнома дар он ба сурати функсия ё class навишта мешавад. Барои эҷоди барнома бояд функсия ё класс мавриди назар навишта шавад. Дар китобхонаҳои стандартии C++ функсияҳо ва class-ҳои зиёде мавҷуданд, ки барнома-соз метавонад аз онҳо низ истифода кунад.

Операторҳои C++ дорои хусусияти ба худ хос мебошанд, ба монанди:

- a) Ҳар оператор дар забони C++ бо аломати «;» ба охир мерасад.
- b) Дарозии ҳар оператор дар як сатр то 255 рамз шуда метавонад.
- c) Як оператор метавонад дар як ё якчанд сатр навишта шавад.
- d) Дар як сатр метавон якчанд операторро навишт.

C++ татбиқи хеле васеъ дорад. Тавассути он бозиҳо, барномаҳои шабакавӣ, 3D-графикҳо, барномаҳои системавӣ, барномаҳои мобилӣ ва ғайра сохта мешаванд.

C++ барои пайдоиши забонҳои Java ва C# заминаи асосӣ ба ҳисоб меравад, зеро аксари операторҳои забонҳои мазкур ба операторҳои C++ монанд мебошанд.

Бояд қайд кард, ки C++ пурра забони C -ро дарбар намегирад. Баъзе воситаҳои дар забони C вучуд доранд, ки барои забони C++ бегона мебошанд. Масалан, C++ даъвати функсияи main-ро дар дохили барнома иҷозат намедиҳад, вале ин амал дар забони C қобили қабул аст.

Алифбои забони C++

Дилхоҳ забони гуфтугӯӣ барои навиштани маълуми худ дорои алифбои хос мебошад. Алифбо имконияти навиштани фикру ақида ва дархостҳои одамонро фароҳам меорад. Забони C++ (дилхоҳ забонҳои барноманависӣ) низ дорои алифбои хос мебошад, яъне барнома аз рамзҳо (харфҳо)-и забон сохта мешавад. Аломатҳои (элементҳои) асосии алифбои забони C++ аз рамзҳои зерин таркиб ёфтаанд:

1. Ҳарфҳои хурд ва калони алифбои латини A, B, C, ..., x, y, z
2. Рақамҳои арабии 0, 1, 2, ..., 9
3. Рамзҳои махсус ~, !, @, #, \$, %, ^, &, *, (), _ , +, {}, ‘, “”, /, \, ?, №, ;, %, :, (), +, , ,
4. Рамзҳои сарҳат, гузаштан ба сатри нав, фосила ва ғайра.

Ҳассосият ба ҳуруфот ва эзоҳ дар забони C++

Забони C++ нисбат ба ҳуруфот (регистр) ҳассос мебошад. Бинобар ин, дар он навишти калимаҳои ABC, Abc, ABc, aBC, AbC, abc аз ҳамдигар тафотути кулӣ доранд. Дар забони мазкур калимаҳои калида (операторҳо, дастурҳо ва фармонҳо) бояд бо ҳарфи хурд навишта шаванд.

Ҳангоми навиштани барнома қитъае аз рамзи (коди) барнома ва ё дастуреро, ки барои хонанда нофаҳмост, метавон эзоҳ дод. Эзоҳ аслан яксатра ва бисёрсатра мешавад. Эзоҳи яксатра бо ду рамзи хати қач (слеш) навишта мешавад. Масалан: // Ин эзоҳи барнома аст.

Ин сатрро барнома сарфи назар мекунад, яъне сатре, ки пас аз // навишта шудааст, хонда намешавад. Эзоҳи бисёрсатра бошад, дар дохили аломати ситорача ва як хати қач (слеш) навишта мешавад. Мисол: /* ин эзоҳи бисёрсатра аст */

Калимаҳои калидии забони C++

Барои ба калимаҳои калидии забони C++ шинос шудан аввал калимаҳои калидии C, сипас калимаҳои калидии C++ -ро дида мебароем. Калимаҳои калидии забони C дар ҷадвали зерин оварда шудаанд:

auto	case	continue
do	else	extern
for	if	long
return	signed	static
switch	union	void
const	volatile	_complex

break	char	default
double	enum	float
goto	int	register
short	sizeof	struct
typedef	unsigned	while
restrict	_bool	_imaginary

Калимаҳои калидии забони C++ ин ҳамаи калимаҳои калидии забони C ва калимаҳои калидии дар қадвали зер овардашуда мебошанд. Аз калимаҳои дар поён овардашуда, калимаи калидии __rtti калимаи калидии махсуси Borland C++ буда, дар лоиҳаи ANSI C++ муайян нагардидааст.

asm	mutable	this
bool	namespace	throw
catch	new	true
class	operator	try
const_cast	private	typeid
delete	protected	typename
dynamic_cast	public	using
explicit	reinterpret_cast	virtual
false	__rtti	wchar_t
friend	static_cast	
inline	template	

Истифодаи калимаҳои калидӣ ба сифати номи тағйирёбандаҳо, функсияҳо ва ғайра мумкин нест.

Идентификаторҳо (identifiers)

Идентификатор гуфта, пайдарпайии ҳарфу рақамҳо ва символҳои махсусро меноманд, ки ҳангоми навиштани барнома барои номгузории объектҳои гуногуни забони барноманависӣ ба монанди: тағйирёбандаҳо, суроғаҳо, функсияҳо, файлҳо ва ғайра истифода бурда мешаванд. Ҳангоми истифодаи идентификатор бояд ба меъёрҳои зерин риоя кард:

➤ Идентификатор бо ҳарфи лотинӣ ва ё симболи махсуси хати поён (_) сар мешавад. Ба мисли: abc, ABC, A328B, a328b, _abs, a, a1, _a, a_b;

➤ Калимаҳои калидӣ ба сифати идентификатор истифода бурда намешаванд;

➤ Дарозии идентификатор набояд аз 32 символ (рамз) зиёд бошад. (Дар стандарти ANSI миқдори символҳои идентификатор номаҳдуд аст. Дар Турбо Си идентификатор то 32 символ шуда метавонад.);

➤ Идентификатор бояд, калимаи пайваст бошад. Ба монанди: My_num, speed_of_body, my_text_file ва ғайра.

Имкониятҳои нави C++

Заминаи асосии пайдоиши C++ ҳарчанд ки забони C ба ҳисоб равад ҳам, вале забони C++ дорои имкониятҳои мебошад, ки дар забон C вучуд надоштанд. Ин имкониятҳо иборатанд аз:

1. Пуштибонӣ аз БОН¹;
2. Пуштибонӣ аз барноманависии умумӣтавассути қолабҳо;
3. Навъи додаҳои нав (bool ва wchar_t);
4. Коркарди ҳолатҳои истисноӣ;
5. Ҳаҳоҳои номӣ;
6. Функсияҳои дохилӣ (inline function);
7. Сарборигузориҳои операторҳо;
8. Сарборигузориҳои функсияҳо;
9. Ишорагарҳо ва операторҳои идоракунии ҳофизаи тавзеъшудаи (парокандашуда) RTT1;
10. Китобхонаҳои стандартии иловагӣ.

Бояд қайд кард, ки имкониятҳои нави C++ ин муаррифӣ намудан ба намуди ифодаҳо, табдилдиҳии навҳо ба намуди функсия, операторҳои new ва delete, навҳои bool ва wchar_t, нишонагарҳо, аргументҳо ба намуди пешфарз, азнавмуаррификунӣ (overriding), ҳаҳоҳои номӣ, классҳо (ба монанди варосат, функсия-аъзо, функсия-ҳои виртуалӣ, классҳои абстрактӣ ва конструктор), азнавмуар-рификунии операторҳо (override operators), шаблонҳо, оператори ::, коркарди ҳолатҳои истисноӣ ва ғайра мебошад.

Дар C++ эзоҳ ба намуди ду хати қач пайдо шуд, ки он аз забони BCPL гирифта шудааст.

Якчанд хусусиятҳои C++ баъдан дар забони C кӯчонида шудаанд. Ба монанди: калимаҳои калидии const, inline, муаррификунӣдар ҳалқаи for, эзоҳ дар услуби C++ (//) ва ғайра.

Компилятор ва интерпритатор

Барномасоз бояд донанд, ки забони барноманависӣ дар компютер хислати барномаро муайян мекунад, на роҳҳои иҷроиши онро. Барои иҷроиши барнома аз транслятор истифода бурда мешавад.

Транслятор (англ. translator), ин барномаи тарҷумон буда, барномаи дар ягон забони барноманависӣ навишташударо ба варианти охири, яъне фармонҳои мошинӣ ва ё коди мошинӣ табдил медиҳад.

Транслятор ба сурати компилятор ё интерпритатор истифода бурда мешавад.

Компилятор (англ. compiler) барнома ё маҷмӯае аз барномаҳои компютерӣ аст, ки матн дар забонҳои барноманависии сатҳи боло навишташударо хонда, онро ба забони сатҳи поён (Target Language), монанди Ассамблер ё забони сатҳи мошинӣ табдил медиҳад. (Дар компилятор амали тарҷумаи барнома якбора иҷро карда мешавад).

Интерпретатор (англ. interpreter — мутарҷим, мутарҷими шифоҳӣ) ба монанди компилятор кор карда, барномаро сатр ба сатр тарҷума ва иҷро мекунад.

Дилхоҳ забони барноманавӣ, вобаста аз мақсади сохтанаш дорои компилятор ё интерпретатор мебошад. Масалан, забони Паскал, ки асосан

¹ БОН- барноманависии ба объектҳо нигаронидашуда

барои ҳалли масъалаҳо пешбинӣ шудааст. Барои он суръати иҷроиш муҳим мебошад, аз ин рӯ, он аз компилятор истифода мекунад, яъне дар он компилятор муайян шудааст.

Забони Бейсик, ки барои навомӯзон сохта шудааст, аз интерпритатор истифода мекунад, яъне дар он барнома сатр ба сатр хонда ва иҷро карда мешавад.

Компиляторҳои озод (бепул)-и C++ ва муҳити интегронидашудан корбар (МИК)

Ба касе пӯшида нест, ки компиляторҳои C++ (аксари барномаҳои компютерӣ) ба ду гурӯҳ тақсим мешаванд:

- ✓ Тиҷоратӣ;
- ✓ Озод.

Интихоби компилятор дар дасти барномасоз мебошад. Касоне, ки пул доранд, метавонанд компилятори дорои калидро гиранд. Ва ашхосе, ки пул надоранд, метавонанд варианти бепули компиляторро интихоб кунанд. Забони C++ забони васеъ буда, дорои компиляторҳои зиёди пулакӣ ва бепул мебошад. Дар поён рӯйхати баъзе компиляторҳои бепули C++ оварда шудаанд:

1. Code::Blocks — яке аз компиляторҳои замонавии IDE мебошад, ки компиляторҳои зиёдеро пуштибонӣ мекунад. Худи компилятори мазкур дар забони C++ навишта шудааст, ки шакли зохирии он қариб ба Microsoft Visual Studio 6 монанд мебошад.

Code::Blocks метавонад, дар системаҳои омили монанди: Windows, Linux ва Mac OS X ва ғайра кор кунад.

Code::Blocks компиляторҳои зиёдеро пуштибонӣ мекунад, ба монанди:

- ✓ MinGW / GCC C/C++
- ✓ GNU ARM GCC Compiler
- ✓ GNU AVR GCC Compiler
- ✓ GNU GCC Compiler for PowerPC
- ✓ GNU GCC Compiler for TriCore
- ✓ Digital Mars C/C++
- ✓ Digital Mars D (бо якчанд маҳдудият)
- ✓ SDCC (Small device C compiler)
- ✓ Microsoft Visual C++ 6
- ✓ Microsoft Visual C++ Toolkit 2003
- ✓ Microsoft Visual C++ 2005-2012 (бо якчанд маҳдудият)
- ✓ Borland C++ 5.5
- ✓ Watcom
- ✓ Intel C++ compiler
- ✓ GNU Fortran
- ✓ GNU ARM
- ✓ GNU GDC

Версияҳои гуногуни Code::Blocks вуҷуд доранд, ки дар айни ҳол аз ҳама версияи наватарини он Code::Blocks 12.11 ба ҳисоб мераванд.

Насби Code::Blocks сода буда, ҳангоми насб ягон чизи иловагиро аз истифодабаранда талаб намекунад.

2. Dev-C++ яке аз компиляторҳои дигари маъмули C++ буда, онро метавон аз суроғайи “<http://www.bloodshed.net/devcpp.html>” ба таври ройгон ба даст овард. Компилятори мазкур дар системҳои омили Windows 95/98/NT/2000/XP/7 метавонад бе мамоният кор кунад. Худи компилятори Dev-C++ дар забони барноманависии Delphi навишта шудааст, ки компиляторҳои оилаи GNU GCC (MinGW, Cygwin)-ро пуштибонӣ мекунад.

Насби Dev-C++ душвор набуда, ҳангоми насб ягон чизи иловагиро аз истифодабаранда талаб намекунад.

Dev-C++ дорои имкониятҳои зиёд мебошад, ба монанди:

- ✓ Мухити графикӣ;
- ✓ Меню бо забони русӣ;
- ✓ Дебогар (debugger)-и дохилӣ GDB;
- ✓ Имконияти сохтани барномаҳои консолӣ ва графикӣ ва ғайра.

3. Microsoft Visual Studio Express- компилятори озод буда, онро метавон аз суроғайи “<http://www.microsoft.com/express/>” ба даст овард. Компилятори мазкур метавонад дар системаҳои омили: Windows XP SP3/Vista/Server 2003/Server 2008/7–кор кунад. Илова бар ин, компилятори MS Visual C++ 2008-2012- ро пуштибонӣ мекунад.

Дар дохили Visual Studio Express воситаҳои зерин ба таври алоҳида вучуд доранд:

- ✓ Visual Basic Express
- ✓ Visual Web Developer Express
- ✓ Visual C++ Express
- ✓ Visual C# Express
- ✓ SQL Server Express

4. Eclipse CDT (C/C++ Development Tooling) - компилятори озод буда, онро метавон аз суроғайи “<http://www.eclipse.org/cdt/>” ба даст овард. Ин компиляторро системаҳои омили: AIX/FreeBSD/HP-UX/Linux/Mac OS X/OpenSolaris/Solaris/QNX/Windows – пуштибонӣ мекунад. Компилятори мазкур компилятори GNU GCC- ро пуштибонӣ мекунад.

5. NetBeans IDE- компилятори озод буда, онро метавон аз суроғайи “<http://www.netbeans.org/>” ба даст овард. Компилятори мазкур метавонад, дар системаҳои омили: FreeBSD/Linux/Mac OS X/ Open-Solaris/Solaris/Windows – кор кунад. Илова бар ин, компилятори GNU GCC- ро пуштибонӣ мекунад.

6. MinGW (англ. Minimalist GNU for Windows) (mingw.org) – компилятори мувофиқ буда, барои навиштани барномаҳои GNU ва Win32 API пешниҳод шудааст. Компилятори мазкур стандарти C99-ро пуштибонӣ карда, дорои ҳамаи пакетҳои лозимӣ мебошад.

7. Open Watcom - компилятори озод буда, онро метавон аз суроғайи “openwatcom.org” ба даст овард. Компилятори мазкур метавонад, дар системаҳои омили: DOS, OS/2, Windows –кор кунад.

8. Digital Mars C/C++ compiler - компилятори озод буда, онро метавон аз суроғаи “digitalmars.com/download/dmcpp.html” ба даст овард. Компилятори мазкур метавонад, дар системаи омили Windows –кор кунад.

9. Borland free compiler - компилятори озод буда, онро метавон аз суроғаи “borland.com/devsupport/bcppbuilder/downloads/” ба даст овард. Компилятори мазкур метавонад, дар системаи омили: Windows 7, Vista, XP –кор кунад. Илова бар ин, компиляторҳои C, C++ ва C# - ро пуштибонӣ мекунад.

10. Turbo C++ - компилятори озод буда, онро метавон аз суроғаи “turboexplorer.com” ба даст овард. Компилятори мазкур метавонад, дар системаи омили: Windows 7, Vista, XP –кор кунад. Илова бар ин компиляторҳои C ва C++- ро пуштибонӣ мекунад.

Муҳити интегронидашудаи коркард (МИК)

МИК ё IDE (Integrated development environment) - системаи воситаҳои барномавӣ мебошад, ки барноманависон онро барои коркарди таъминоти барномавӣ истифода мебаранд.

МИК дорои воситаҳои зерин мебошад:

1. Редактори матнӣ
2. Компилятор / интерпретатор
3. Воситаҳои автоматикунонӣ
4. Дебогар (debugger)

Бисёр МИК-ҳои ҳозиразамон дорои мурургари (браузери) классҳо, равзанаи дарахтшакли объектҳо ва диаграммаи дарахтшакли классҳо (барои БОН) мебошанд. МИК-и забонҳои Eclipse, Embarcadero RAD Studio, Qt Creator, NetBeans, Xcode ва Microsoft Visual Studio, Borland Delphi чунин аст.

Равзанаи тахриргари коди лоиҳа ё сохтори барнома

Тавре ки қаблан зикр шуд, ин равзана ба чор қисмат тақсим мешавад ва дар ҳар қисмат амали лозимӣ анҷом дода мешавад.

Дар қисмати аввал китобхона (header files) насб карда мешавад. Китобхона файлест, ки дар он маълумоти зарурӣ оид ба ягон функсия ва ё фармони забон нигоҳ дошта мешавад. Масалан, барои иҷрои амалҳои математикӣ аз библиотекаи <math>, барои дохилкунию хориҷкунӣ аз библиотекаи <iostream> ва ғайра истифода бурда мешаванд.

Дар блоки муаррифӣ намудани class-ҳо ва функсияҳо бошад, class-ҳо ва функсияҳо таъриф карда мешаванд. Барои ҳалли як масъалаи бузург барномасоз метавонад, онро ба қисматҳои бо ҳамдигар вобаста ҷудо кунад, ки ҳар яке аз он қисматҳоро як функсия номида мешавад. class-ҳо ҳам монанди функсияҳо барои як барномаро ба қисматҳо тақсим кардан истифода бурда мешаванд. Як класс метавонад функсияҳоро дар дохилаш нигоҳ дорад. Баъдтар ин матолибро мавриди баҳс қарор медиҳем.

Методи асосӣ ё сарфунксия функсияест, ки барнома қорашро аз он оғоз мекунад, яъне, дилхоҳ барнома, ки дар забони C++ навишта мешавад, дорои ақалан як функсия (main) аст, ки барнома қорашро аз он ҷо оғоз мекунад. Маҳз ин қисматро қисми оғози барнома мегӯянд.

Қисмати охирон бошад, блоки эзоҳи функцияҳо ва ё коди асосӣ номида шуда, дар дохили қавсҳои фигуравӣ навишта мешавад. Махсус дар ин қисмат фармонҳои асосии барнома навишта мешаванд. Барои пурратар шинос шудан ба ин матлаб дар поён қисматҳои асосии барнома овардашудаанд.

<pre> } } int main() { return 0; } </pre>	<pre> //Қисмати аввал, яъне қисми муаррифи намудани китобхонаҳо //Блоки муаррифӣ намудани class-ҳо, функцияҳо ва пешнамунаҳои функцияҳо // Методи асосии барнома (main) // Блоки эзоҳи функцияҳо ва ё коди асосӣ. </pre>
---	---

Тавре ки аз ин нақша дида мешавад, дар қисмати аввал библиотекаҳо пайваст карда мешаванд. Ҳар як объект ва функцияҳои стандартии забони С++ дар ҳамон ҳолате кор меkunанд, ки агар библиотекаи ба он вобаста пайваст карда шавад. Масалан, барои дохил ва хориҷ кардани маълумот библиотекаи `iostream` пайваст карда мешавад.

Қисмати дуввум қисматест, ки дар он функцияҳо `class`-ҳо ва сохторҳо муаррифӣ карда мешаванд. Оянда дар мавзӯи функцияҳо ва `class`-ҳо дар ин бора тавзеҳот хоҳем дод.

Қисмати саввум ин функцияи асосии барнома мебошад, ки аз он ҷо коди барнома оғоз мегардад. Ҳар барнома дар забони С++ ҳатман дорои як функция мебошад, ки онро функцияи асосӣ мегӯянд. Қисмати охирон қисматест, ки дар он коди барнома навишта мешавад. Намунаи барнома дар С++.

<pre> #include <iostream> using namespace std; int main() { cout<<"Варнамаи наҳустини ман дар забони С++"; cin.get(); return 0; } </pre>	<pre> } Библиотекаҳои набкардашуда } Коди барнома </pre>
--	---

Дар мавзӯи функцияҳои дохилкунӣ ва хориҷкунӣ дар ин бора тавзеҳоти бештаре хоҳем дод.

Китобхонаҳои (Библиотекаҳои) С++

Забони С++ забонест, ки аз якҷанд қисматҳои бо ҳамдигар вобаста ташкил ёфтааст. Далели ин гуфтаҳо китобхонаҳои С++ мебошанд. Ҳангоми бо забони С++ кор кардан бояд библиотекаи лозимӣ дар барнома насб карда шавад. Библиотекаҳо дар диски мантиқие, ки дар он барномаи С++ насб карда шудааст, дар дохили парвандаи `include` мавҷуданд. Наҳви насб намудани як китобхона шакли зеринро дорад:

`#include <номи_китобхона>`

Дар версияҳои куҳнаи C++ пас аз номи ҳар китобхона ҳарфи h навишта мешуд, ки он ҳарфи аввалаи калимаи header (header file-сар файл) мебошад. Навиштани ҳарфи “.h” ҳамаи объектҳои библиотекарро фаъол мегардонад (яъне ба кор омода месозад). Дар сурати навиштани “.h” бояд дар пеши ҳар як объект ҳангоми истифодабарӣ яке аз операторҳои зерин навишта шавад:

`std::номи_объект ё using std::номи_объект`

Сатри аввала дар пеши ҳар як объект дар сурати истифодашави навишта мешавад. Сатри дуюм бошад, барои ҳар як обект як маротиба пас аз муаррифи намудани китобхона навишта мешавад.

Илова бар ин, дар стандарти нави C++ метавон барои ҳамаи библиотекаҳо як маротиба дастури зеринро навишт. `using namespace std;`

Ин оператор ҳамаи объектҳои библиотекаҳоро фаъол мегардонад.

Дар забони C++ Тавре ки қайд кардем, библиотекаҳои зиёде мавҷуданд, ки дар ҷадвали зерин якҷандтои онҳо оварда шудаанд:

№	Номи библиотека	Маъно ва мафҳум
1	<code>iostream</code>	Китобхонаи дохилкунӣ ва хориҷкунӣ чараёнҳо
2	<code>complex</code>	Китобхонаи кор кардан бо классҳои <code>complex</code>
3	<code>math</code>	Китобхонаи математикӣ
4	<code>string</code>	Китобхонаи кор бо сатрҳо
5	<code>stdio</code>	Китобхонаи стандартии дохилкунӣ ва хориҷкунӣ
6	<code>stdlib</code>	Китобхонаи стандартӣ
7	<code>iomanip</code>	Китобхонаи кор бо монипулятори додаҳо
8	<code>graphics</code>	Китобхонаи кор бо муҳити графикӣ

Қайд. Барои якуҷабора пайвасти кардани ҳамаи китобхонаҳои лозими аз дастури зерин истифода бурда мешавад. `#include <bits/stdc++.h>`

Саволҳо барои мустаҳкамкунӣ

- 1) Забони C++ нахустин маротиба кай ва аз тарафи кӣ сохта шудааст?
- 2) Хусусиятҳои забони C++ кадомҳоянд?
- 3) Кадом рамзҳо ба алифбои C++ дохил мешаванд?
- 4) Компилятор аз интерпритатор чӣ фарқ дорад?
- 5) Кадом компиляторҳои C++ ро медонед?
- 6) Чӣ тавр C++ ба кор омода карда мешавад?
- 7) Созандаи C++ чӣ маслиҳат додааст?
- 8) Чӣ тавр дар забони C++ барои барнома эзоҳ навиштан мумкин аст?
- 9) Эзоҳи як сатра аз бисёрсатра чӣ фарқ дорад?
- 10) Кадом рамз барои эзоҳи бисёрсатра ва яксатра истифода бурда мешаванд?
- 11) Китобхонаи `iostream` барои чӣ пешбинӣ шудааст ва шакли кутоҳшудаи кадом калимаҳо мебошад?
- 12) Барои кор кардан бо функсияҳои математикӣ аз кадом китобхона истифода бурда мешавад?

Боби 2. Додаҳо, тағйирёбандаҳо ва амалҳо

Муқаддима

Барои ҳалли ягон масъала дар дилхоҳ барнома бояд муайян бошад, ки додаҳо (data)–и он масъала чӣ гунаанд ва ба он додаҳо чӣ амал иҷро карда мешавад. Дар забони барномасозии С++ ду намуди додаҳо мавҷуданд: доимӣ ва тағйирёбанда. Додаҳои доимӣ додаҳоеанд, ки дар тамоми тӯли барнома қимати худро тағйир намедиханд, аммо додаҳои тағйирёбанда бошанд, баракс амал мекунанд, яъне қимати худро метавонанд дар дилхоҳ ҷои барнома тағйир диҳанд, қимати нав қабул кунанд ва ғайра. Додаҳо метавонанд аз навъ (тип ё type)-ҳои гуногун иборат бошанд. Ба монанди бутун, ҳақиқӣ (касрӣ), рамзӣ сатрӣ ва ғайра. Ба ғайр аз ин, додаҳо маҳаллӣ, умумӣ ва минтақавӣ мешаванд. Пеш аз истифодабарӣ додаҳо бояд муаррифӣ карда шаванд. Дар забони С/С++ додаҳоро метавон дар дилхоҳ ҷойи барнома муаррифӣ кард. Дар вақти муаррифӣ пеш аз номи дода навъи (типи) он нишон дода мешавад. Навъи муаррифӣ намудани як дода ба сурати зерин аст :

Навъи_дода номи_дода;

Навъи додаҳо дар С/С++ ба сурати зерин дастабандӣ карда мешаванд:

1) Додаҳо аз навъи бутун (integer), ки аз ҳашт навъ иборат мебошанд.
2) Додаҳо аз навъи касрӣ ё ҳақиқӣ (floating-points), ки он ҳам се навъ аст.
3) Додаҳои бутун ва доимӣ (integer constants) -ин намуди додаҳоро метавон ба ғайр аз намуди муқаррарӣ ба намудҳои даҳӣ (decimal), ҳаштӣ (octal) ва шонздаҳӣ (hexadecimal) навишт.

4) Додаҳо аз навъи ишорагар (pointer type).

5) Додаҳо аз навъи шумориши (enumerated type).

6) Додаҳо аз навъи арифметикӣ (arithmetic type) -додаҳои бутун ва ҳақиқӣ дар якҷоягӣ.

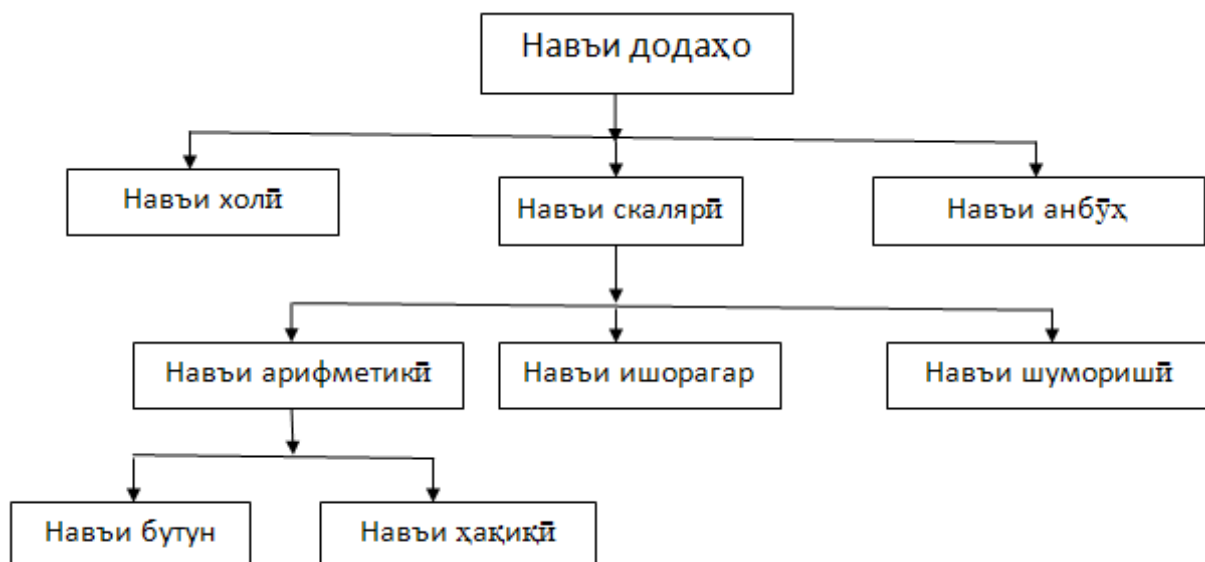
7) Додаҳо аз навъи скалярӣ (scalar type) -додаҳои ишорагар, шуморишӣ ва арифметикиро дар маҷмӯъ ташкил медиҳанд. Аз сабаби он ки ин навъ додаҳо бо додаҳои навъи худ қобили муқоиса ҳастанд, бинобар ин, онҳоро скалярӣ мегӯянд. Яъне барои онҳо операторҳои муқоиса ҷой доранд.

8) Додаҳо аз навъи анбӯҳ (aggregate type) ба ин гурӯҳ дохил мешаванд: массивҳо, сохторҳо (structures) ва қисмҳо (union).

Ин навъ додаҳо барои сохтани тағйирёбандаҳое, ки ба таври мантиқӣ ба ҳамдигар алоқаманданд, истифода бурда мешаванд. Дар мавзӯи сохторҳо оид ба ин навъ додаҳо пурратар эзоҳ медиҳем.

9) Додаҳо аз навъи ҳолӣ (void) – ин навъ додаҳо аслан функция (function)-ҳое, ҳастанд, ки қимат барнамегардонанд.

Ба намуди гистограмма метавонем додаҳои забони С/С++-ро ба намуди зерин тасвир кард:



Додаҳои навъи (типи) рамзӣ

Додаҳои навъи рамзӣ ба таври пешфарз дар хотира як байт ҷойро ишғол мекунанд. Додаҳои ин навъ 255 рамзи ASCII-код буда, метавонанд бо аломат ва ё беаломат бошанд. Қимати додаҳои боаломат дар фосилаи -128 то +127 ҷойгир буда, онҳоро unsigned char мегӯянд. Навъи рамзии бе аломат бошад, signed char номида шуда, фосилаи қиматҳояш аз 0 то 255 мебошад.

Номи навъ	Фосилаи қиматҳо	Андоза ё дарозии навъ
char	-128..127	1 байт
unsigned char	0...255	1 байт
signed char	-128...127	1 байт

Дар аксар забонҳои барномасозӣ додаҳои бутун аз рамзи фарқ мекунанд, яъне 5 ҳамчун як адади бутун ва “a” ҳамчун як рамз фаҳмида мешавад. Вале агар мо дақиқтар нигарем, рамзҳо низ адади бутун мебошанд, яъне ҳар як рамз аз рӯи кодгузориҳои стандарти ASCII (American Standard Code for Information Interchange) ва ebsd (extended binary-coded decimal interchange code) дорои коди муайян аст. Дар забони C++ ҳар як рамз ҳамчун адади бутун фаҳмида мешавад. Одатан рамзҳо дар дохили нохунаки якто-якто навишта мешаванд.

Додаҳои навъи бутун

Додаҳои навъи бутун ба додаҳои аломатдор ва беаломат ҷудо мешаванд. Ҳангоми муаррифӣ намудани додаҳои бо аломат пеш аз навъи дода калимаи «signed» (signed-аломат) ва додаҳои беаломат «unsigned» (unsigned-бе аломат) навишта мешавад. Додаҳои навъи мазкур ҳангоми муаррифӣ намудан дар хотира 2 ва ё 4 байтро ишғол мекунанд.

Додаҳои беаломат барои ҳалли масъалаҳое, ки дар онҳо натиҷа мусбат нест, ба монанди ҳаҷм ва масоҳати ҷисмҳо, суръат ва ғайра истифода бурда мешавад.

Дар чадвали зерин ҳамаи навъҳои бутун ва фосилаи қиматҳояшон оварда шудаанд.

Навъи додаҳо	Фосилаи қиматҳо	Андоза
int	-2147483648...2147483647	4 байт
unsigned int	0...4294967295	4 байт
signed int	-2147483648...2147483647	4 байт
short int	-32768...32767	2 байт
long int	-2147483648...2147483647	4 байт
unsigned short int	0...65535	2 байт
signed short int	-32768...32767	2 байт
long long int	$-(2^{63}-1)...(2^{63}-1)$	8 байт
unsigned long int	0...42949667295	4 байт
signed long int	-2147483648...2147483647	4 байт
unsigned long long int	$0...2^{64}-1$	8 байт

Додаҳои навъи ҳақиқӣ

Додаҳои навъи бутун наметавонанд ҳамаи талаботи моро қонеъ гардонанд. Аз ин хотир, истифодаи додаҳои навъи ҳақиқӣ зарурат мешавад. Додаҳои навъи ҳақиқиро метавонем ба ду тарз истифода барем. Тарзи якум намуди муқаррарӣ, яъне байни қисми бутун ва қисми адади ҳақиқӣ аломати вергул (дар забони C++ нукта (dot)) гузошта мешавад. Масалан, 2.7, 54.471, 0,0025 ва ғайра.

Равиши дуюм, ки намоиши илмӣ (scientific notation) низ номида мешавад. Як кӯтоҳнависӣ дар қолаби дараҷагӣ (экспонентсиалӣ) буда, дар ин навъи навишт адад ба ду қисм ҷудо карда мешавад: қисми монтис (montica) ва қисми дараҷа (exponent). Қисми дараҷагӣ одатан пас аз қисми монтис меояд. Байни ин ду қисмат ҳарфи E навишта мешавад, ки он ҳарфи аввалаи калимаи Exponent - дараҷа мебошад. Қисми дараҷагӣ бояд дар нишондиханда (ё дараҷа)-и 10 омада, ба монтис зарб шавад. Наҳви умумии ин равиш шакли зеринро дорад:

$$mE \pm p$$

Дар ин ҷо m-монтиса(қисми бутуни адад), E-ишораи дараҷа ва p бошад, дараҷа ё қисми дараҷа мебошад. Ба мисолҳои зерин тавачҷух фармоед.

$$1) 0,25E+2=0,25*10^2=0,25*100=25$$

$$2) 3E4=3*10^4=3*10000=30000$$

$$3) 254E - 2 = 254 * 10^{-2} = 254 * \frac{1}{10^2} = 254 * \frac{1}{100} = 2,54$$

Бояд қайд кард, ки қимати дараҷа (намо ҳам гуфта мешавад) наметавонад адади ғайрибутун бошад.

Додаҳои ин навъ float, double ва long double мебошанд. Ба намуди пешфарз додаҳои навъи float чор байтро дар хотира банд мекунанд. Дар намуди дараҷагӣ то 23 разряд дар қисми монтис ва 8 разряд дар қисми намо ё дараҷа қабул карда метавонанд. Додаҳои навъи double бошанд, тӯлашон (дарозияшон) ду баробари додаҳои навъи float мебошад, яъне онҳо дар хотира $2*4=8$ байт ҷойро ишғол мекунанд. Дар баъзе компиляторҳои C++ додаҳои ин навъро

метавон ба сурати long float муаррифӣ кард. Аз ин чо навиштҳои зерин бо ҳамдигар эквивалент мебошанд. double a,b,c,d; ⇔ long float a,b,c,d;

Барои додаҳои навъи double дар намуди дараҷагӣ то 52 разряд қисми мантиса ва 11 разряд қисми дараҷа ё намо шуда метавонад. Агар пеш аз навъи double калимаи long нависем, фосилаи қиматҳои ин тип 10 байт афзоиш мекунад. Дар ҷадвали зерин ҳамаи навъҳои ҳақиқӣ оварда шудаанд.

Навъи додаҳо	Фосилаи қиматҳо	Андоза
float	3.4E-38...3.4E+38	4 байт
double	1.7E-308...1.7E+308	8 байт
long double	3.4E-4932...3.4E+4932	8 байт

Додаҳои навъи мантиқӣ

Додаҳои навъи мантиқӣ яъне bool танҳо ду қиматро метавонанд қабул кунанд, true (ҳақ) ва false (ноҳақ). Дар забони C++ ҳар қимате, ки ғайринулӣ аст, ҳамчун true (ҳақ) фаҳмида шуда, ҳар гуна қимати нулӣ false (ноҳақ) фаҳмида мешавад. Бо ифодаи дигар қимати ҳақ ба як баробар буда, қимати ноҳақ ба нол баробар аст.

Додаҳои навъи ҳолӣ

Ин навъ додаҳо дар дигар стандартҳо мавҷуд набуданд ва танҳо дар стандарти ANSI илова карда шуданд. Ин навъ додаҳо дар функсияҳо, ки қимат барнамегардонанд ва танҳо амали муайянеро иҷро мекунанд, истифода бурда мешаванд. Дар мавзӯи функсияҳо ба ин матлаб равшании зиёд меандозем.

Тағйирёбандаҳо ва муаррифӣ намудани онҳо (variables declaration)

Тағйирёбанда гуфта, бузургии (додае)-ро меноманд, ки дар та-моми тӯли барнома метавонад, қиматҳои гуногун қабул кунад. Тағйирёбандаҳо барои коркарди додаҳо истифода шуда, бо ёрии номашон муайян карда мешаванд, яъне номи тағйирёбанда-суроға (адреси) онро дар хотира нишон медиҳад. Номи тағйирёбанда бояд бо ҳарф сар шавад ва наметавонад калимаи калидии забон бошад, яъне истифодаи калимаҳои калидии забонро мумкин нест ба сифати номи тағйирёбанда истифода кард.

Ҳангоми истифода бурдани тағйирёбандаҳо дар барнома аввал онҳо бояд муаррифӣ карда шаванд, яъне муайян шаванд, ки онҳо ба кадом навъ таалуқанд ва чӣ қадар хотираро банд мекунанд.

Дар аксар забонҳои барноманависӣ тағйирёбандае, ки навъаш муайян нашудааст, ба компилятор фаҳмо намебошад, яъне компилятор ҳангоми тарҷума кардани барномаи навишташуда он тағйирёбандаро намефаҳмад ва дар натиҷа, амали тарҷума иҷро намегардад. Наҳви муаррифӣ намудани тағйирёбанда ба сурати зерин аст:

```
data_type Variable1, Variable2, ..., Variablen
```

Дар ин чо data_type-навъи тағйирёбанда (яъне дилхоҳ навъи стандартӣ), Variable1, Variable2, ..., Variablen-номи тағйирёбандаҳои яқум, дуҷум ва ҳоказо

n-ум мебошанд. Тавре ки дида мешавад, номи тағйирёбандаҳо бо рамзи «,» аз ҳамдигар ҷудо карда мешаванд.

Шояд дар зехни хонандаи мо боз саволе пайдо шавад, ки чӣ ниёзест аз муаррифӣ намудани тағйирёбанда ва он дар барнома-нависи чи аҳамияте дорад? Метавон ба ин савол чунин ҷавоб гуфт, ки ҳангоми муаррифӣ намудани тағйирёбанда ба компилятор гуфта ме-шавад, ки ин тағйирёбанда аз қадом навъ аст, вобаста ба навъи он дар ҳофиза ҷоё муайян банд мешавад. Ҳофиза ба воситаи байтҳо пур карда мешавад. Барои дарозии навъи додашударо муайян кардан метавон аз функсияи sizeof() истифода кард. Ин функсияро баъдтар мавриди баҳс қарор медиҳем. Дар ҷадвали зерин ном ва дарозии навъҳои стандартӣ гирд оварда шудаанд.

Номи навъҳои стандартӣ	Дарозии навъи додашуда бо байт	Фосилаи қиматҳои навъи додашуда
bool	1	Қимати ҳақ (true ё 1) ва ноҳақ (false ё 0)
char	8	аз -128 то 127
char_t	16	аз 0 то 65536
doubl	64	аз 2.2E-308 то 1.8E+308
float	32	аз 1.8E-39 то 1.8E+38
int	32	аз -2147483648 то 2147483647
unsigned char	8	аз 0 то 255
signed char	8	аз -128 то 127
unsigned int	32	аз 0 то 4294967295
signed int	32	аз -2147483648 то 2147483647
short int	16	аз -32768 то 32767
unsigned short int	16	аз 0 то 65536
signed short int	16	аз -32768 то 32767
long int	32	аз -2147483648 то 2147483647
unsigned long int	32	аз 0 то 4294967295
igned long int	32	аз -2147483648 то 2147483647
long double	64	аз 1.8E-38 то 1.8E+38

Асосан дар забони C/C++ навъи додаҳои зерин истифода бурда мешаванд.

1. char —рамзӣ
2. int-бутун
3. float-қасрӣ
4. double-дуқарата
5. bool-мантиқӣ
6. wchar_t-рамзии навъаш ду байта
7. void-холӣ

Барои қолиббанди кардан ба дигар навъҳо аз асоси навъҳои, ки спесификатор ном доранд, истифода карда мешавад. Дар забони C++ чор спесификатори навъи додаҳо мавҷуд аст.

1. short-кўтоҳ
2. long-дароз
3. signed-нишонадор ё аломатдор
4. unsigned-бенишона ё беаломат

Дар чадвали боло баъзе типҳоро мо дидем, ки пеш аз номи онҳо калимаи signed ё unsigned навишта шуда буд. Ин маънои аломатдор ё беаломат будани тағйирёбандаҳои вобастаи ин типҳоро нишон ме-диҳад.

Пас аз дониستاني ин маълумотҳо биёед наҳви муаррифӣ намудани тағйирёбандаҳоро аз нав ба таври возеҳтар дида бароем. Тавре ки қаблан гуфтем, пеш аз тағйирёбанда навъи он бо ҳарфи хурд нишон дода мешавад. Масалан, бигзор тағйирёбандаи *x* аз навъи бутун бошад, пас, барои муаррифӣ намудани он дар забони C++ аз равиши зерин истифода мебарем: `int x;`

Тавре ки дидем, пеш аз номи тағйирёбанда навъи он навишта шудааст. Барои муаррифӣ кардани якчанд тағйирёбандае, ки аз ҷиҳати навъ якхела мебошанд, як маротиба навиштани навъ кифогӣ мекунад. Қолаби умумии ин равиш шакли зеринро дорад:

`data_type var1, var2, ..., varn`

Дар ин ҷо `data_type` навъии додасуда, `var1, var2, ..., varn`-бошанд, тағйирёбандаҳои яқум, дуум ва ҳоказо *n*-ум мебошанд. Ба ғайр аз ин, метавон ҳар як тағйирёбандаро алоҳида-алоҳида муаррифӣ кард. Ба монанди :

`data_type var1; data_type var2; ...; data_type varn`

Дар ин ҷо барои ҳар як тағйирёбанда алоҳида муаррифӣ шудааст.

Тавре ки айён аст, равиши аввала хело сода ва осон мебошад. Масалан бигзор тағйирёбандаҳои *a, b, c, d, e, f* аз навъи бутун бошанд. Онҳоро ба намуди зерин муаррифӣ намудан мумкин аст:

`int a, b, c, d, e, f; ё int a; int b; int c; int d; int e; int f;`

Метавон калимаҳои калидиро ҳангоми яке аз ҳафи онҳоро калон навишта ба сифати номи тағйирёбанда истифода бурд. Мисол:

`int Int;`

Дар ин ҷо тағйирёбандаи `Int` аз навъи `int` эълон шудааст, вале мас-лиҳат он аст, ки аз ин равиш истифода бурда нашавад.

Барои муаррифӣ намудани тағйирёбанда бояд ба қоидаҳои зерин риоя кард:

1. Номи навъи додасуда бояд ба ҳарфи хурд навишта шавад.
2. Номи тағйирёбанда маънидор буда, ба барномаи сохтасуда му-вофиқат кунад.
3. Номи тағйирёбанда бояд ҳатман ба ҳарф ё рамзи хати поён сар шавад.
4. Номи тағйирёбанда бояд калимаҳои калидии забон набошад.
5. Барои навъҳои гуногун бояд номҳои гуногун истифода шавад, яъне бо як ном дар як барнома ду тағйирёбанда ҳарчанд, ки аз навъҳои гуногунанд муаррифӣ нагардад. Ба таври дигар, ду тағйирёбанда набояд ҳамном бошанд.

Қимати аввала бахшидан ба тағйирёбандаҳо

Ҳангоми муаррифӣ намудани тағйирёбандаҳо метавонем ба онҳо якуякбора қимати аввала бахшид. Бигзор, тағйирёбандаҳои a, b, c, d аз навъи бутун буда, мувофиқан ба 10, 20, 14, 17 баробар бошанд. Ҳангоми муаррифӣ намудани онҳо метавон ин қиматҳоро ба тағйирёбандаҳои мазкур ба сифати қимати аввала бахшед (яъне тағйирёбандаҳоро инитсиализатсия кунем).

`int a=10, b=20, c=14, d=17;`

Ба ҳамин монанд метавон ба тағйирёбандаҳои дигар типҳо низ қимати аввала бахшид.

Аз як навъ ба навъи дигар табдил додани тағйирёбандаҳо

Забони C/ C++ иҷозат медиҳад, ки додаҳои як навъро ба навъи дигар табдил дода шаванд. Барои иҷрои ин амал, яъне тағйирёбандаи як навъро ба навъи дигар табдил додан, тағйирёбандаи лозимро дар дохили қавс ва пас аз қавс номи тағйирёбандаи мавриди назарро менависем. Масалан, бигзор тағйирёбандаи a аз навъи `int` бошад, барои онро ба навъи `float`-и бо номи b табдил додан амали зеринро иҷро мекунем, яъне:

`int a=14; b= (float) a`

Ин навишт пеш аз он ки қимати бутуни a -ро ба b бахшад, онро ба навъи `float` табдил медиҳад. Тавре ки дидем, оператори `cast` танҳо як қимат қабул мекунад ҳалос, аз ин рӯ, онро оператори ягона (би-нари)-и дорои як операнда меноманд. Ин равиш дар бисёр ҳолатҳо хеле муфид аст. Масалан, бигзор тағйирёбандаҳои x ва y аз навъи бутун буда, дорои қиматҳои мувофиқан ба 10 ва 4 баробар бошанд ва тағйирёбандаи z аз навъи `float` бошад. Ифодаи зерини $z=x/y$ ба 2.0 баробар мешавад, яъне ин ифода танҳо қисми бутуни натиҷаро намоиш медиҳад. Барои он ки қисми бутун ва касрии ададро якбора намоиш диҳад, бояд аққалан яке аз тағйирёбандаҳо x ва y низ аз навъи `float` бошанд ва ϵ бо ϵ оператори `cast` якешон ба навъи `float` табдил дода шавад, яъне амали зерин иҷро гардад.

`z=(float)x/y`

Дар ин равиш натиҷаи $z=2.5$ хоҳад шуд ва агар ин мисолро ба намудҳои зерин низ табдил диҳем, натиҷаи $z=2,5$ хоҳад шуд.

`z=x/(float)y` ё `z=(float)x/(float)y`

Аз ин мисолҳо метавонем хулоса баровард, ки дар дилхоҳ ҷойи барнома метавонем додаи навъии дилхоҳи стандартиро ба навъи дигар табдил диҳем, яъне амали `casting` метавонад дар дилхоҳ ҷои барнома иҷро шавад. Наҳви умумии ин равиш ба сурати зерин аст:

`(data-type) expression`

Мисол. Бигзор тағйирёбандаҳои a ва b аз навъи `double` бошанд, ифодаи зерин суммаи ин тағйирёбандаҳоро ба намуди бутун табдил медиҳад.

`y= int (a+b)`

Ин ифода аввал қиматҳои a ва b -ро бо ҳамдигар ҷамъ намуда, баъд натиҷаашро ба навъи бутун гардонидани ба тағйирёбандаи y мебахшад. Ба ҳамин монанд метавон амали `casting`-ро барои дилхоҳ типҳои стандартӣ истифода кард.

Доимиҳо (constant)

Дар математика баъзе бузургҳои ҳастанд, ки қимати доимӣ доранд ба монанди $\pi \approx 3,14$, $e \approx 2,718$ ва ғайра. Дар барноманависӣ низ ин равиш истифода бурда мешаванд. Пас, доимиҳо бузургҳои ҳастанд, ки дар тамоми тӯли барнома қимати худро тағйир намеди-ҳанд, яъне баръакс тағйирёбандаҳо мебошанд. Наҳви муаррифӣ намудани доимиҳо ба сурати зил аст:

```
const data-type variable=value ё const номи доими =қимати доимӣ
```

Дар ин ҷо const-шакли кӯтоҳи constant-доимӣ буда, value қимати доимӣ ва variable бошад, номи доимӣ мебошад. Доимӣ метавонад аз дилхоҳ навъи базавӣ (стандартӣ) бошад, ба монанди бутун, ҳақиқӣ, рамзӣ сатри ва ғайра. Масалан, бузургии доимиро бо понздаҳ рақам баъди вергул метавонем ба сурати зерин муаррифӣ кунем:

```
const double Pi=3.141592653589793;
```

Ба ғайр аз ин, метавонем доимиҳоро бо ёрии пешпардозанда (препроцессор)-и #define пеш аз функсияи асосӣ (main) муаррифӣ кунем. Дар ин сурат номи доими бояд бо ҳарфҳои калон навишта шавад.

```
#define PI 3.141592653589793
```

Тавре ки мушоҳида мешавад, ҳангоми бо ёрии пешпардо-занда муаррифӣ намудани доимӣ навъи он нишон дода намешавад. Дар ин ҳолат дар тӯли барнома доимиро метавонем ҳамчун адади ҳақиқии дароз истифода барем. Дар мавзӯи пешпардозанда (препроцессор)-ро пурратар баҳс мекунем.

Амалҳо

Тағйирёбандаҳо, доимиҳо, массивҳо, сохторҳо ва ё class-ҳо бо истифода аз амалҳо бо ҳамдигар пайваст карда шуда, ифодаҳо ташкил медиҳанд. Аз ин ҷо амалҳоро бо сурати зерин таъриф медиҳем.

Таъриф. Амалҳо аломат ё нишонаҳои ҳастанд, ки барои сохтани (эҷод намудани) ифодаҳо хизмат мекунанд.

Баъзан ба ҷои калимаи амал, аз амалгар низ истифода карда мешавад. Амалҳоро метавонем ба сурати зерин дастабандӣ кард:

- 1) Амалҳои арифметикӣ (arithmetic operators)
- 2) Амалҳои мантикӣ (logical operators)
- 3) Амалҳои муқоисакунӣ (relation operators)
- 4) Амали шартӣ (conditional operator)
- 5) Амали коммо (comma) ё вергул
- 6) Амалҳои ягона (unary operators)
- 7) Амалҳои ҳофиза ()

Амалҳои арифметикӣ (arithmetic operators)

Амалҳои арифметикӣ амалҳои мебошанд, ки барои иҷрокунии амалҳои арифметикӣ ба додаҳо истифода бурда мешаванд. Дар ҷадвали зерин амалҳои мазкур ҳам оварда шудаанд.

Номи амал	Нишона Нишона(symbol)-и	Истифодабарии Намунаи амалҳои	Истифодабарии Намунаи амалҳои
-----------	----------------------------	----------------------------------	----------------------------------

	амал	арифметикӣ дар математика	арифметикӣ дар забони C++
Чамъ	+	$a + b$	$a + b$
Тарҳ	-	$a - b$	$a - b$
Тақсим	/	$\frac{a}{b}$	a/b
Зарб	*	ab	$a * b$
Бақия аз тақсим	%	$a \text{ mod } b$	$a \% b$

Ҳамаи ин амалҳо ба ғайр аз амали бақия аз тақсим (%), дар тамоми забонҳои барноманависӣ як хел истифода бурда мешаванд. Оператори % -ро remainder operator ё modulus operators низ мегӯянд, монанди дигар забонҳои барноманависӣ истифода бурда мешаванд, яъне бақияи тақсими a-ро бо b муайян мекунад. Ба ғайр аз ин, амалҳои зерин мавҷуданд.

Номи амал	Нишона (symbol)-и амал	Намунаи амал дар математика	Намунаи амал дар забони C++
Чам ягона	+	$a = a + b$	$a += b$
Тарҳи ягона	-	$a = a - b$	$a -= b$
Як воҳид афзоиш (increment)	++	$a = a + 1$	$a ++ \text{ ё } ++ a$
Як воҳид коҳиш (decrement)	--	$a = a - 1$	$a -- \text{ ё } -- a$
Бахшиш	=	$a = b$	$a = b$

Ду амали ++ ва -- дар дигар забонҳои барномасозӣ вучуд надорад. Ин амалҳо адади додашударо як воҳид зиёд ва як воҳид кам мекунанд. Ҳар кадоме аз ин амалҳо дар ду намуд ё қолаб истифода бурда мешаванд. Тарзи истифодаи онҳо дар ҷадвали зерин оварда шудааст.

№	Амали афзоиш (++)	Мисол	Амали коҳиш (--)	Мисол
1	Пешафзоиш (preincrement)	$++ a$	Пешкоҳиш (predecrement)	$-- a$
2	Пасафзоиш (postincrement)	$a ++$	Паскоҳиш (postdecrement)	$a --$

Амали бахшиш (default assignment operator) бошад, баробари математикӣ набуда, балки қимати b-ро ба a мебахшад.

Қайд. Фарқи асосии байни амали пешафзоиш (пешкоҳиш) ва пасафзоиш (паскоҳиш) дар он аст, ки амали пешафзоиш (пешкоҳиш) пеш аз иҷрои дигар амалҳо қимати амалвандашро як воҳид афзоиш (коҳиш) медиҳад. Амали пасафзоиш (паскоҳиш) бошад, аввал амалҳои лозимиро иҷро карда, сипас қимати амалвандашро як воҳид афзоиш (коҳиш) медиҳад.

Мисол. Бигзор $a=5$ бошад онгоҳ қимати ифодаи $y=+++a+4$ ба 10 баробар буда, қимати ифодаи $y=4+a++$ ($y=a+++4$) ба 9 баробар мешавад/

Амалҳои кор бо битҳо

Дар забони C++ якчанд амале мавҷуд аст, ки онҳоро амалҳои кор бо битҳо мегӯянд. Сабаби амали бо бит гуфтани онҳо дар он аст, ки онҳо аввал қимат (адад)-и додасударо ба намуди дӯй фарз намуда, баъд бо он амалиётеро иҷро мекунанд. Ин амалҳо иборатанд аз:

1. Амали ва (&);
2. Амали ё (!);
3. Амали хориҷкуни ё (^);
4. Амали тарҳ ~ ;
5. Кӯчиш ба рост ($A \ll B$);
6. Кӯчиш ба чап ($A \gg B$).

1) Амали ва (&) –амали кор бо битҳо (хусусан системаи дӯй) буда, чунин амал мекунад, дар сурати ҳам қимати якум ва ҳам дуюмаш ҳақ будан ба қимати ҳақ доро буда, дар дигар сурат ба қимати ноҳақ доро аст, яъне:

x	y	$x \& y$
1	1	1
0	1	0
1	0	0
0	0	0

Мисоли 1. Бигзор $x=25$ ва $y=45$ бошад ва онҳоро ба намуди системаи дӯй гардонем, чунин шаклро мегиранд:

X	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
Y	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	1
$x \& y$	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

Аз ин ҷо натиҷаи $x \& y = 25 \& 45 = 9$ мешавад.

2) Амали ё (!)-амали кор бо битҳо буда, ба қимати ҳақ дорост, агар ақалан яке аз қиматҳои додасуда ҳақ бошанд, яъне

x	y	$x y$
1	1	1
0	1	1
1	0	1
0	0	0

Мисоли 2. Мисоли 1-ро барои ин амал дида мебароем, яъне бигзор $x=25$ ва $y=45$ бошад ва онҳоро ба намуди системаи дӯй гардонем, чунин шаклро мегиранд:

x	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
y	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	1
$x y$	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1

Пас, аз ин ҷо натиҷаи $x | y = 25 | 45 = 61$ мешавад.

3) Амали ё (^)-амали кор бо битҳо буда, ба қимати ҳақ доро аст, агар танҳо яке аз қиматҳои додасуда ҳақ бошанд, яъне

x	y	$x \wedge y$
1	1	0

0	1	1
1	0	1
0	0	0

Мисоли 3. Мисоли 1-ро барои ин амал дида мебароем, яъне бигзор $x=25$ ва $y=45$ бошад ва онҳоро ба намуди системаи дӯи гардонем, чунин шакл-ро мегиранд:

x	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
y	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	
x^y	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	

Пас, аз ин ҷо натиҷаи $x^y=25^45=52$ мешавад.

4) Амали тарҳ \sim -амали кор бо битҳо буда, ба қимати ҳақ доро аст, агар қимати додашуда ноҳақ бошанд ё баракс, яъне

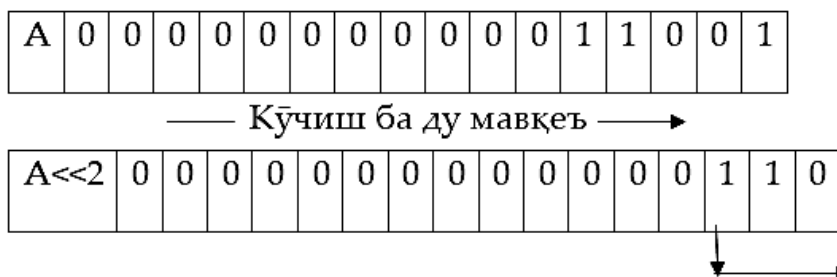
x	1	0	1	0
$\sim x$	0	1	0	1

5) Кӯчиш ба чап ($A \ll B$) –ин амал адади A -ро дар намуди дӯи фарз намуда, баъд онро B разряд ё мавқеъ ба тарафи чап мекӯҷонад. Ми-сол: бигзор $A=25$ бошад ва $B=2$ бошад, дар аввал A -ро ба намуди дӯи мегардонад, яъне



Аз ин ҷо $A \ll B = 25 \ll 2 = 100$

6) Кӯчиш ба рост ($A \gg B$) ин амал адади A -ро дар намуди дӯи фарз намуда, баъд онро B разряд ё мавқеъ ба тарафи рост мекӯҷонад. Мисол: бигзор $A=25$ бошад ва $B=2$ бошад, дар аввал A -ро ба намуди дӯи мегардонад, яъне



Аз ин ҷо $A \gg B = 25 \gg 2 = 5$

Амалҳои муқоисакунӣ (relation operators)

Амалҳои муқоисакунӣ барои дар байни ду тағйирёбанда муносибат ё робита барқарор намудан хизмат мекунанд, яъне муайян мекунанд, ки дар байни ин тағйирёбандаҳо чӣ гуна муносибат мавҷуд аст. Дар ҷадвали зерин амалҳои муқоисакунӣ оварда шудаанд.

Номи амал	Нишона (symbol)-и амал дар математика	Нишона (symbol)-и амал дар C++	Истифодабарии амалҳои муқоисақунӣ дар забони C++	Қимати амал дар забони C++
Калонтар	>	>	a>b	a калон аз b
Хурдтар	<	<	a<b	a хурди b
Калон ё баробар	≥	>=	a>=b	a калон ё баробари b
Хурд ё баробар	≤	<=	a<=b	a хурд ё баробари b
Баробар	=	==	a==b	a айнан баробар b
Нобаробар	≠	!=	a!=b	a нобаробари b

Барои амалҳои муқоисақунӣ ва мантиқӣ калимаҳои калидӣ номи true ва false истифода бурда мешаванд. Дар забони C++ ҳар гуна адади ғайринули ҳамчун true ва адади нулӣ бошад, ҳамчун false фаҳмида мешавад.

Амалҳои мантиқӣ (logical operators)

Амалҳои мантиқӣ амалҳои ҳастанд, ки барои иҷрокунии амале болои ифодаҳои мантиқӣ истифода бурда шуда, дорои қимати ҳақ (true) ва ноҳақ (false) мебошанд. Дар қадвали зерин операторҳои мантиқӣ оварда шудаанд.

№	Номи амал	Аломат (symbol)-и амал дар забони C++	Аломат (symbol)-и амал дар математика	Номи амал дар математика
1	and-ва	&&	∩	Конъюнксия
2	or-ё		∪	Дизъюнксия
3	not-не	!	~	Инкор

Тавре ки қайд кардем, дар забони C++ қимати true ба 1 баробар буда, қимати false ба 0 баробар мешавад. Ин амалҳо дар алгебраи Булл чунин таъриф дода мешаванд:

2) Конъюнксияи мулоҳизаҳои A ва B гуфта, ҳамин гуна мулоҳизаи C-ро меноманд, ки ба қимати ҳақ (true) доро аст, агар ҳам мулоҳизаи A ва ҳам мулоҳизаи B баробари true, яъне ҳақ бошанд. Дар акси ҳол, дорои қимати ноҳақ (false) мешаванд.

Мулоҳизаҳои яқум	Мулоҳизаҳои дуюм	Мулоҳизаи яқум && мулоҳизаи дуюм
False	False	False
False	True	False
True	False	False
True	True	True

2) Дизъюнксияи мулоҳизаҳои A ва B гуфта, ҳамин гуна мулоҳизаи C-ро меноманд, ки ба қимати ҳақ (true) доро аст, агар аққалан яке аз мулоҳизаҳои A

ва ё В ба қимати true, яъне ҳақ доро бошанд. Дар акси ҳол, дорои қимати ноҳақ (false) мешаванд.

Мулоҳизаи якум	Мулоҳизаи дуюм	Мулоҳизаи якум мулоҳизаи дуюм
False	False	False
False	True	True
True	False	True
True	True	True

3) Инкори мулоҳизаи А гуфта, мулоҳизаи \tilde{A} -ро меноманд, ки он ба қимати ҳақ дорост, агар ҳуди мулоҳизаи А ба қимати ҳақ доро бо-шад, дар акси ҳол, ба қимати ноҳақ баробар мешавад.

№	Мулоҳиза	Инкори мулоҳиза
1	False	True
2	True	False

Мисол. Бигзор тағйирёбандаҳои a,b,c,d аз навъи бутун буда, мувофиқан ба 2, 4, c, d баробар бошанд. Акнун дар ҷадвали зерин ҷанд ифодаеро меорем.

№	Мулоҳизаҳо	Натиҷаи мантиқӣ	Натиҷа дар забони C++
1	(a<5) && (x>0)	true	1
2	((2*a+3)>=c) && (a>c)	false	0
3	(a==2) (a<(d+5))	true	1
4	(b/a)==a	true	1
5	(a<b) (a<4)	false	0

Қайд. Баъзан ба сифати дуто амперсанд (&&) аз якто & низ истифода бурда мешавад. Фарқият дар он аст, ки амали дуто амперсанд дар ҳолати ҷой надоштани шартӣ якум (false будани амалванди якум) шартӣ (амалванди) дуюмро тамоман тафтиш намекунад. Амали якто амперсанд бошад, новобста аз шартӣ (амалванди) якум шартӣ (амалванди) дуюмро низ тафтиш мекунад.

Мисол. Бигзор ададҳои бутуни $a=5$ ва $b=4$ дода шуда бошанд, онгоҳ пас аз иҷрои дастури `bool y=(++a>6) && (b++>3)` қимати $a=6$ ва $b=4$ ва ҳангоми иҷрои дастури `bool y=(++a>6) & (b++>3)` қимати $a=6$ ва $b=5$ мешавад.

Амали шартӣ (conditional operator) ё сегона (тернерӣ)

Амали шартӣ барои санҷидани дурустӣ ва ё нодурустии шартӣ додасуда истифода бурда мешаванд. Амали мазкур дорои се операнд мебошад, яъне

Номи амал	Нишона (symbol)-и амал	Наҳви амал	Амали иҷромекардаи амал
Шартӣ	?:	Exp1?exp2:exp3	exp1 шартӣ гузошташуда буда, дар сурати ғай-ринулӣ ё дуруст буда-ни он exp2 иҷро мегардад, дар ҳолати акс exp3 иҷро мегардад.

Ин амал амали амали if else-ро иҷро мекунад.

If a>0 then

Z=25
Else
Z=-25

Масалан, қисми барномаи боло, ки дар забони Visual Basic навишта шудааст. Метавонем онро бо ёрии амали шартӣ дар забони C++ ба сурати зерин нависем:

Z=((a>0)?25:-25);

Ба ҳамин монанд метавонем дигар мисолҳоро низ иҷро кунем.

Амали коммо (The comma operator) ё вергул

Амали коммо ё вергул имконияти иҷрошавии якчанд амалиётро дар дохили як оператор мебахсад. Ин амал бештар дар операторҳои даврии for истифода бурда мешавад. Наҳв ва истифодаи он дар қадвали зерин оварда шудааст.

Номи амал	Нишона (symbol)-и амал	Наҳви амал	Амали иҷромекардаи амал
Коммо	,	exp1,exp2,...,expn	Аввал exp1 баъд exp2 ва ҳоказо expn аз назар гузаронида мешаванд.

Тарзи навишти он ба сурати зерин аст:

variable1=(var1,var2,...varn);

Мисол: x=(y=5,y+25)

Ин ифода аввал ба тағйирёбандаи у қимати 5-ро бахшида, баъд натиҷаро ба 25 ҷамъ мекунад ва натиҷаи ҳисобшударо ба а мебахсад. Дар натиҷа а ба қимати 30 баробар мешавад. Ин амал асосан дар дохили сикли for истифода бурда мешавад баъдтар дар мавзӯҳои оянда мисолҳои мушаххасро бо ёрии ин оператор меорем.

Амалҳои ягона (Unary operators)

Амалҳои ягона амалҳоеанд, ки фақат дар рӯи як операнд амал мекунанд. Ин амалҳо дар қадвали зерин гирд оварда шудаанд.

№	Номи амал	Рамз (symbol)-и амал	Истифодаи амал дар забони C++
1	Ҷамъи унарӣ (unary plus)	+	a+=5
2	Тарҳи унарӣ (unary minus)	-	a-=2
3	Амали навъ (cast operator)	cast	(data_type) variable
4	Амали андоза (sizeof)	sizeof	sizeof(datatype)

Амали ҷамъи унарии a+=5 ба ифодаи a=a+5 ва тарҳи унарии a-=5 ба a=a-5 эквивалент мебошад. Амали cast барои навъи додашударо ба дигар навъи базавӣ табдил додан истифода мешавад. Амали sizeof –амале мебошад, ки барои муайянкунани бузургии операнди худ ба ҳисоби байт истифода бурда мешавад. Амалванди (операнди) ин амал ҳамаи навъҳои стандартӣ мебошанд, яъне ин амал андозаи навъҳои базавиро муайян мекунад. Наҳви умумии ин амал тamenст:

variable =sizeof(data-types)

Мисол: a=sizeof(int); b=sizeof(double) ва ғайра.

Амалҳои ҳофиза

Дар забони C/C++ якчанд амалҳои мавҷуданд, ки адрес ё маҳалли физики тағйирёбандаи додшуда (нишондодашуда)-ро метавонад ба барномасоз дастрас гардонад. Дар ҷадвали зерин ин амалҳо оварда шудаанд.

№	Номи амал	Аломат	Наҳви амал	Амали иҷрокардагӣ
1	Суроға (адрес)	&	&a	Суроғаи тағйирёбандаи a-ро нишон медиҳад.
2	Нишона (ссылка ё deference)	*	*a	Қимати додае, ки суроғаи он дар тағйирёбандаи a захира шудааст, бармегардонад
3	Элементи массив	[]	a[n]	Элементи (n+1)-уми массивро муайян мекунад
4	Нуқта (dot)	.	a.b	Қимати тағйирёбанда (ё элемент)-и b, ки аз сохтори a мебошад, бармегардонад.
5	Ақрабак ба рост (right arrow)	->	a-> b	Қимати тағйирёбанда (ё элемент)-и b, ки суроғаи он дар тағйирёбандаи ишорагари a қарор дорад, бармегардонад.

Авлавияти амалҳо

Дилхоҳ ифода дар забонҳои барноманависӣ аз операндҳо (тағйирёбандаҳо, доимиҳо ва ғайра) ташкил карда мешавад. Аз ин рӯ ҳангоми навиштани як ифода, зарурати донишҷӯ авлавияти амалҳо пеш меояд, яъне лозим мешавад то бидонем, кадом амал аввал иҷро мешавад. Бигзор ифодаи зерин дода шуда бошад:

$$y=++3+6/2++$$

барои муайян кардани қимати ин ифода, лозим аст, ки тартиби иҷроиши амалҳоро донем, то ин ки қимати матлабро ба даст ояд. Дар ҷадвали зерин тартиби иҷроиши амалҳо оварда шудаанд.

Аввалият	Амал	Маъно ва мафҳум	Равиши амал
1.Баландтарин	::	Иҷозати ҳудуди фаъолият	Аз чап ба рост Left to Right
2	++	Як воҳид афзоиш (намуди пасвандӣ)	
	--	Як воҳид коҳиш (намуди пасвандӣ)	
	()	Даъвати функсия	
	[]	Дастрасӣ ба элементи массив	

	.	Интихоби элемент аз рӯи ишора	
	->	Интихоби элемент аз рӯи нишона	
	const_cast	Табдилдиҳии навъ (const cast) (C++)	
	dynamic_cast	Табдилдиҳии навъ (dynamic cast) (C++)	
	reinterpret_cast	Табдилдиҳии навъ (reinterpret cast) (C++)	
	static_cast	Табдилдиҳии навъ (static cast) (C++)	
3	++	Як воҳид афзоиш (намуди пешвандӣ)	Аз чап ба рост (Left to Right)
	--	Як воҳид коҳиш (намуди пешвандӣ)	
	+	Ҷамъи унарӣ	
	-	Фарқи унарӣ	
	!	Инкори мантиқӣ	
	~	Инкори битӣ	
	(type)	Қолаб	
	*	Гирифтани ишорагар	
	&	Гирифтани суроға	
	sizeof	Size-of (андоза)	
	new, new[]	Ҷудо намудани ҳофизаи динамикӣ (C++)	
	delete, delete[]	Озод намудани ҳофизаи динамикӣ (C++)	
4	.* ->*	Ишорагар ба аъзо (C++) Ишорагар ба аъзо (C++)	Аз чап ба рост (Left to Right)
5	* / %	Зарб Тақсим Бақияи тақсим	
6	+ -	Ҷамъ Тарҳ	
7	<< >>	Лағжиши бити ба чап Лағжиши бити ба рост	
8	< <= > >=	Хурд Хурд ё баробар Калон Калон ё баробар	

9	= !=	Баробар Нобаробар		
10	&	Ва –и битӣ		
11	^	Ё-и битии истисно (XOR)		
12		Ё-и битӣ		
13	&&	Ва-и мантиқӣ		
14		Ё-и мантиқӣ		
15	?:	Амали шартӣ		Аз рост ба чап (Right to Left)
16	=	Баҳшиш		Аз рост ба чап(Right to Left)
	+=	Баҳшиш бо ҷамъ		
	-=	Баҳшиш бо тарҳ		
	*=	Баҳшиш бо зарб		
	/=	Баҳшиш бо тақсим		
	%=	Баҳшиш бо бақия аз тақсим		
	<<=	Баҳшиш бо кӯчиш ба чап		
	>>=	Баҳшиш бо кӯчиш ба Рост		
	&=	Баҳшиш бо ва-и Мантиқӣ		
16	^=	Баҳшиш бо ё-и Мантиқии битӣ		
	=	Баҳшиш бо ё-и битӣ		
17	Throw	Оператори коркарди ҳолтҳои истисноӣ (C++)		
18	,	Амали Comma	Аз чап ба рост (Left to Right)	

Қайд: Аксари ин амалҳо дар рафти китоб эзоҳ дода мешаванд.

Тавре ки аз ин ҷадвал дида мешавад, дар мисоли овардашуда, аввал амали як воҳид афзоиш ба намуди пасвандӣ, баъд амали як воҳид афзоиш ба намуди пешвандӣ, баъд амали тақсим ва дар охир амали ҷамъ иҷро карда мешавад. Аз ин ҷо қимати у=6 мешавад.

Таъриф. Тартиби иҷроиши амалҳоро, авлабияти амалҳо меноманд.

Қайд. Амалҳо метавонанд, аз чап ба рост ва ё баракс иҷро шаванд.

Муродифи амалҳо

Дар забони C++ калимаҳои калидӣ муаррифӣ шудаанд, ки ба якҷанд амалҳо муродиф мебошанд. Ба монанди: and (&&), bitand (&), and_eq (&=), or (||), bitor (|), or_eq (|=), xor (^), xor_eq (^=), not (!), not_eq (!=), compl (~). Амалкарди онҳо бо амалҳо айнан монанд буда, ба сифати амалҳо метавон онҳоро истифода кард, яъне онҳо варианти дигари амалҳо мебошанд. Масалан, амали bitand на танҳо барои иваз намудани амали битии ва (&), инчунин

метавонад, ба сифати амали суроға низ истифода шавад. Масалан, операторҳои зерин бо ҳам баробарқувва мебошанд:

$$\text{int bitand ref} = n \Leftrightarrow \text{int \& ref} = n$$

Дар стандарти ANSI C (ISO C) ин калимаҳои калидӣ, ҳамчун макросҳои пешпардозанда муайян карда шудаанд, ки дар китобхонаи <iso646.h> нигоҳ дошта мешаванд. Барои якпорчагӣ бо C++ бошад, китобхонаи <ciso646> мавҷуд аст.

Ифодаҳо (expression)

Ифодаҳо маҷӯи маънидоре мебошанд, ки бо ёрии амалҳои арифметикӣ, мантикӣ, муқоисанамой ва ғайра ташкил карда мешаванд.

Ифодаҳоро ба сурати зерин метавон дастабандӣ кард:

1. Ифодаҳои арифметикӣ
 2. Ифодаҳои муқоисавӣ
 3. Ифодаҳои мантикӣ
- 2) Ифодаҳои арифметикӣ ифодаҳои ҳастанд, тасе ёрии амалҳои арифметикӣ аз рӯи қоидаи хосе ташкил карда шудаанд. **Мисол.**

№	Ифодаҳои риёзӣ	Ифодаҳои забони C++
1	$(a+b)c$	$(a+b)*c$
2	$\frac{a}{b+c}$	$a/(b+c)$

Дар мисоли аввал адади a бо адади b чамъ намуда, сипас натиҷа бо c зарб карда мешавад. Дар мисоли дуввум бошад, аввал b бо c чамъ карда мешавад, сипас адади a бо натиҷа тақсим карда мешавад.

2) Ифодаҳои муқоисакунӣ ифодаҳои ҳастанд, тасе ёрии амалҳои муқоисакунӣ бо риояи қонуни махсусе ташкил карда мешаванд.

Қимати ифодаҳои муқоисакунӣ ҳақ ва ё ноҳақ шуда метавонад, яъне агар шартҳои гузашташуда дуруст бошанд, қимати ифодаи муқоисакунӣ ҳақ ва дар сурати акс, ба қимати ноҳақ баробар мегардад. Дар аксари забонҳои барномасозӣ қимати ҳақ-ро true ва қимати ноҳақро false мегӯянд. Дар забони C++ бошад, қимати ҳақ ба 1 (ё ба адади ғайринулии калон аз нол) баробар буда, қимати ноҳақ ба 0 баробар аст.

№	Ифодаҳои риёзӣ	Ифодаҳои забони C++
1	$a > b$	$a > b$
2	$(a+b) \geq c$	$(a+b) \geq c$

3) Ифодаҳои мантикӣ ифодаҳои ҳастанд, ки аз ифодаҳои арифметикӣ ва муқоисакунӣ аққалан бо як амали мантикӣ ташкил шудаанд. Нисбат ба ду ифодаҳои пешин ин навъ ифодаҳо мураккабтар ва пе-чидатар мебошанд. Қимати ифодаҳои мантикӣ ҳам ба ҳақ ё ноҳақ баробар шуда метавонанд.

№	Ифодаҳои риёзӣ	Ифодаҳои забони C++
1	$a > b > c$	$(a > b) \&\& (b > c)$
2	$\frac{x}{y} \leq \frac{a}{b} < 2xy$	$((x/y) \leq (a/b)) \&\& ((a/b) < 2*x*y)$
3	$a \leq b$	$(a < b) \ \ (a == b)$

Операторҳо

Операторҳо ҳукмҳои ҳастанд, ки сабаби иҷро кардани амале мегарданд. Дар забони C++ операторҳои зерин истифода бурда мешаванд:

1) Операторҳои сода операторҳоеанд, ки бо ёрии онҳо амалҳои ғайришартӣ иҷро карда мешаванд. Масалан, операторҳои зерин сода мебошанд:

- а) бахшиш;
- б) хондан ва навиштан ё дохилкунӣ ва хоричкунӣ;
- в) даъват (фурӯхони)-и функсия;
- г) Гузариши бешарт ё равон кардани барнома ба нуктаи нишондодашуда, яъне goto;

2) Операторҳои сохта операторҳоеанд, ки аз навъи сохторҳои алгоритмӣ таркиб ёфтаанд, ба монанди:

- а) Операторҳои мураккаб (compound statement) аз тарафе ё зиёда операторҳои сода таркиб ёфтаанд;
- б) Операторҳои даврӣ (repetitive statement) операторҳое ҳастанд, ки аз алгоритмҳои даврӣ таркиб ёфтаанд;
- в) Операторҳои шартӣ (conditional tamen) операторҳое ҳастанд, ки аз алгоритмҳои шартӣ таркиб ёфтаанд.

Дар оянда ин операторҳоро ба таври пурра шарҳ медиҳем.

Функсияҳои стандартӣ дар забони C++

Забони C++ монанди дигар забонҳо аз функсияҳои стандартии зиёде истифода мебарад. Аксари ин функсияҳоро функсияҳои мате-матикӣ низ мегӯянд ва дар сурате онҳо ба кор мебароянд, ки агар ки-тобхонаи <math.h> пайваст карда шуда бошад. Дар ҷадвали зерин рӯйхати ин функсияҳо оварда шудааст.

№	математика	C + +	Маъно ва мафҳум
1	$ x $	<i>abs(x)</i>	Қимати мутлақи адади x-ро муайян мекунад.
2	-	<i>fabs(x)</i>	Қимати мутлақи адади ҳақиқии x-ро муайян мекунад.
3	<i>sinx</i>	<i>sin(x)</i>	Синуси адади x-ро муайян мекунад.
4	<i>cosx</i>	<i>cos(x)</i>	Косинуси адади x-ро муайян мекунад.
5	<i>tgx</i>	<i>tan(x)</i>	Тангенс адади x-ро муайян мекунад.
6	<i>atgx</i>	<i>atan(x)</i>	Аргтангенс адади x-ро муайян мекунад.
6	<i>asinx</i>	<i>asin(x)</i>	Аркасинуси адади x-ро муайян мекунад.
7	e^x	<i>exp(x)</i>	Экспоненти адади x-ро муайян мекунад.
8	<i>logx</i>	<i>log(x)</i>	Қимати логарифми натуралии адади x-ро муайян мекунад ($x > 0$)
9	<i>lgx</i>	<i>log10(x)</i>	Қимати логарифми даҳии адади x-ро муайян мекунад ($x > 0$)
10	\sqrt{x}	<i>sqrt(x)</i>	Қимати решаи квадратии адади x-ро муайян мекунад ($x \geq 0$)
11	a^x	<i>pow(a, x)</i>	Адади a-ро ба дараҷаи x-мебардорад
12	-	<i>ceil</i>	Адади ҳақиқиро то калон шудан аз қисми

			бутунаш яклухт мекунад
13	-	<i>modf</i>	Адади додашударо ба қисми бутун ва касрӣ чудо мекунад
14	-	<i>floor</i>	Адади ҳақиқиро то аз қисми бутунаш хурд шудан яклухт мекунад
15	-	<i>fmod</i> ё %	Бақияи тақсими адади x -ро бар y муайян мекунад

Ҳамаи ин функцияҳо қимати ҳақиқӣ қабул мекунанд. Баъзан функцияи $\text{pow}(a,x)$ қимати нодуруст медиҳад ё қимате намедиҳад, яъне дар ҳолате ки адади a мусбат буда, x адади касрӣ бошад, дар он ҳолат ин функция қимат дода наметавонад. Пас, барои дар ин ҳолат қимат гирифтани онро ба намуди зерин менависем. $\text{pow}(\text{fabs}(a), x)$

Дар забони C++ баъзе аз ин функцияҳо барои дилхоҳ навъ сарборӣ карда шудааст, масалан барои онки оператори $\text{pow}(a,x)$ болои ададҳои бутуни дароз амалеро анҷом диҳад, зарур аст, ки ба сурати $\text{powl}(a,x)$ навишта шавад. Дар ин ҷо l –ҳарфи аввали калимаи `long` мебошад.

Саволҳо барои мустаҳкамкунӣ

1. Дода гуфта, чиро дар назар доранд?
2. Додаҳои навъи бутун аз додаҳои рамзи (`char`) чӣ фарқ доранд?
3. Тағйирёбанда чист ва чӣ тавр муаррифӣ карда мешавад?
4. Амали `casting` чӣ гуна амал аст?
5. Чӣ тавр ба тағйирёбанда қимати аввала бахшидан мумкин аст?
6. Оператор чист ва он барои чӣ лозим аст?
7. Операторҳо чанд хел мешаванд?
8. Ҳар яке аз операторҳоро эзоҳ диҳед?
9. Ифодаҳо барои чӣ лозиманд?
10. Ифодаҳои мантиқӣ аз муқоисакунӣ чӣ фарқ мекунанд?
11. Оператор чист ва операторҳо чанд хел мешаванд?
12. Операторҳои сода аз мураккаб чӣ фарқ доранд?
13. Функцияи `abs` аз `fabs` чӣ фарқ дорад?
14. Решаи дараҷаи n дар забони C++ чӣ тавр навишта мешавад?
15. Оё метавон дар дилхоҳ ҷойи барнома тағйирёбандаҳои лозимиро муаррифӣ кард?
16. Барои кадом амалҳо муродиф мавҷуд аст?
17. Чи гуна амалҳоро амалҳои ягона меноманд?
18. Амали `коммо` чи гуна амал аст?
19. Амалҳои ҳофиза чӣ гуна амаланд?
20. Додаҳои навъи ҳоли кадом вақт истифода бурда мешаванд?
21. Доимӣ чист ва чи тавр муаррифӣ карда мешавад?
22. Функцияи `floor` аз функцияи `ceil` чи фарқ мекунад?
23. Функцияи `fmod` барои чӣ пешбинӣ шудааст?

Масъалаҳо барои кори мустақила

Ифодаҳои зеринро дар забони C++ нависед.

1	$y = x^5 - 2(x + a)$	2	$y = x^{x^2} + 2^{x^2}$
3	$y = x - 2 (x + 2)$	4	$y = 0.5^{\frac{2x^2 - 6x}{x - 3}}$
5	$y = \frac{x - 1}{ x - 1 } (x^2 - 4)$	6	$y = \left \log_2 \frac{x - 4}{x^2 - 4} \right $
7	$y = \sqrt{10^{\lg x^2}}$	8	$y = \frac{x}{ x } \sin 2x$
9	$y = x^2 - 7x + 6 $	10	$y = 2^{\sqrt{-\sin^2 x}}$
11	$y = x ^2 - 7 x + 6 $	12	$y = 5^{\frac{1}{3} \log_5(x-1)}$
13	$y = 2x^2 + 3x - 1$	14	$y = x - x^3$
15	$y = x^2 - 8\sqrt{x} - 32$	16	$y = 2x^2 - 3x + 1$
17	$y = \arcsin x + 2x^2$	18	$y = 2x^2 + 3$
19	$y = 2x + \ln \sin x + 2 \cos x $	20	$y = x \cdot (\sin \ln x - \cos \ln x)$
21	$y = \operatorname{arctg}(\operatorname{sh} x) + (\operatorname{sh} x) \cdot \ln \operatorname{ch} x$	22	$y = e^x \cdot (\cos 2x + 2 \sin 2x)$
23	$y = e^x \cdot (\cos 2x + 2 \sin 2x)$	24	$y = x \cdot (\sin \ln x - \cos \ln x)$
25	$y = \sqrt{x} - (1 + x) \cdot \operatorname{arctg} \sqrt{x}$	26	$y = x \cdot (\sin x^2 - \cos \ln x)$
27	$y = \log_{\sin x} \cos x$	28	$y = \cos[\cos(\cos x)]$
29	$y = (\cos x)^{\sin x}$	30	$y = \operatorname{tg}(\sin(x + y))$

Боби 3. Воситаҳои дохилкуниву хориҷкунӣ дар C++

Муқаддима

Барои ягон маълумотро ба компютер дохил кардан ва ё маълумотро ба чоп додан дар забони C++ функцияҳои (functions) махсусе истифода мешаванд. Функцияҳои дохилкуниву хориҷкунӣ дар китобхонаи (library)-и «iostream» чойгиранд, ва барои аз ин функсияҳо (объектҳо) истифода бурдан дар аввал бояд ин китобхона ба барнома пайваст карда шавад. Яъне дар блоки пайвасти китобхонаҳо бояд фармонҳои зерин навишта шаванд:

```
#include <iostream>
using namespace std;
```

Дар ин ҷо фармони #include пешпардозанда буда, iostream тахфифи се калима input-дохилкунӣ, output-хориҷкунӣ ва stream- ча-раён буда, дар яқҷоягӣ маънои чараёни дохилкуниву хориҷкуниро дорад. Ҳарфи h- бошад, аз ҳарфи аввали калимаи header-гирифта шуда, маънои сарлавҳаро дорад. Ҳангоми ин библиотекарро пайваст намудан функцияҳои (объектҳои) cin, cout ва ғайра ба кор медароянд.

Барои маълумоти хориҷ кардашударо қолиббанди намудан бояд аз функцияҳои библиотекаи iomanip истифода бурда шавад, ин библиотека низ монанди дигар библиотекаи пеш аз истифодабарии пайваст карда мешавад.

Функцияҳои дохилкуниву хориҷкунӣ хеле ба регистр (register) ҳассос мебошанд. Аз ин ҷо бояд навъи тағйирёбандаҳое, ки дохил ё хориҷ карда мешаванд, муайян бошад, то ки ба таври автоматӣ коркард гарданд, вагарна барнома ба хатогӣ рубарӯ мешавад.

Ба ғайр аз функцияҳое, ки то ҳол номбар шуданд, забони C++ метавонад аз функцияҳои дохилкуниву хориҷкунии C истифода кунад. Функцияҳои асосии дохилкуниву хориҷкунӣ дар забони C scanf() ва printf() мебошанд, ки онҳоро функцияҳои қолабдор мегӯянд. Бо ёрии ин функцияҳо ададҳо, рамзҳо ва сатрҳо ба компютер (барнома) дохил ва хориҷ карда мешаванд. Ба ғайр аз ин, функцияҳои getchar() ва putchar() барои дохилкунӣ ва хориҷкунии як рамз ва функцияҳои gets() ва puts() барои дохилкуниву хориҷкунии сатрҳо истифода бурда мешаванд. Барои аз ин функцияҳо истифода намудан бояд библиотекаи stdio пайваст карда шавад. stdio низ тахфифи се калимаи англисии standard-стандарт, input-дохилкуни ва output-хориҷкунӣ буда, маънояш дохилкуниву хориҷкунии стандартӣ мебошад.

Функцияҳои дохилкуниву хориҷкуни метавонанд дар дилхоҳ ҷои барнома истифода шаванд. Ҳоло ба навбат функцияҳои дохилкунӣ ва хориҷкунии C ва баъд C++-ро дида мебароем.

Функцияҳои дохилкуниву хориҷкунии қолабдор (formatted console input/output)

Тавре ки гуфтем, дар забони C функцияҳои scanf() ва printf() функцияҳои қолабдор мебошанд, яъне ин функцияҳо ба барномасоз имконият медиҳанд, ки додаҳои гуногунро ба барнома дохил ва хориҷ намуда, коркард кунад. Функцияи printf() барои додаҳои гуногунро дар саҳифаи тасвир (console) чоп намудан истифода бурда мешавад, функцияи scanf() бошад, баръакси printf()

амал мекунад, яъне додаҳоро аз саҳифаи вурудӣ меҳонад. Ҳар яке аз ин функсияҳоро ба таври муфассал дида мебароем.

Функсияи printf()

Функсияи printf() барои чоп намудани дилхоҳ навъи додаҳо истифода бурда мешавад. Ин функсия метавонад якчандто аргумент дошта бошад, ки онҳо аз ҳамдигар бо вергул ҷудо карда мешаванд. Аргументҳои сатрӣ дар дохили ноҳунак навишта мешаванд ва Тавре ки гуфтем, ин функсия функсияи қолабдор мебошад ва қолаби додаҳои он дар дохили қавс навишта мешаванд. Наҳви умумии истифодабарии ин функсия ба намуди зерин аст:

```
printf("control string ",arg1,arg2,...,argn)
```

Дар ин ҷо control string-сатр ё рамзи идоракунанда буда, arg1, arg2, ..., argn аргументҳои ин функсия мебошанд.

Сатрҳои идоракуни чунин сатрҳои мебошанд, ки дар дохили ноҳунак (double quotation) навишта шуда, навъи аргументҳои функсияро муайян мекунанд. Дар ҷадвали зерин якчандтои онҳо оварда шудаанд.

Код	Маънои фармони қолаб(навъи иттилооти чопшаванда)
%c	(character data Item)- Додаҳои навъи рамзӣ
%d	(Signed decimal Integer)-Додаҳои навъи бутуни аломатдор
%f	(floating-point decimal notation)-Додаҳои навъи ҳақиқӣ
%e	(floating-point exponential notation)-Додаҳои навъи floating_point - и аломатдор
%g	(floating-point (%f or %e) width char shorter)-Додаҳои навъи floating-point бо қолаби %f ё %e
%u	(Unsigned decimal integer)-Додаҳои навъи бутуни беаломат
%s	(String)-Додаҳои сатрӣ
%o	(unsigned octal integer)-Додаҳои навъи бутун дар асоси системаи ҳисоби ҳаштӣ
%x	(unsigned hexadecimal integer)- Додаҳои навъи бутун дар системаи ҳисоби шонздаҳӣ
%L	(Prefix used with %d, %u, %x, %o, to specify long integer)-Адади бутуни дароз дар қолаби %d, %u, %x, %o
%I	(Signed decimal integer)-Додаҳои навъи бутуни аломатдор
%p	(display a pointer)-Додаҳои навъи ишорагар

Мисоли 1. Истифодаи функсияи printf().

```
#include <iostream>
using namespace std;
int main(){
    double Pi = 3.14159;
    int Int = 25;
    printf("%f %d", Pi, Int);
    cin.get();
    return 0;
}
```

Дар ин чо қолаби %d –нишондиҳандаи адади бутун (яъне якум аргументи функцияи printf () адади бутун мебошад), фармати %f-нишондиҳандаи ададҳои ҳақиқӣ (яъне нишон медиҳад, ки аргументи дуҷуми функцияи printf () адади ҳақиқӣ мебошад). Як фосилаи ҳолӣ байни қолабҳо гузоштем барои натиҷаҳоро аз ҳамдигар бо як ҷои ҳолӣ ҷудо кардан. Агар ҷои ҳолӣ гузошта нашавад натиҷаи дуввум аз паси натиҷаи аввала меояд ва дидани натиҷаҳои ҷори имконнопазир мегардад.

Аргументҳои функцияи мазкур на танҳо ададҳо (тағйирёбандаҳо), балки метавонанд ягон ифода бошанд.

Мисоли 2. Хориҷкунии ифодаҳо ва ададҳо.

```
#include <iostream>
using namespace std;
int main(){
    double Pi = 3.14159;
    int Int = 25;
    printf("%f %d", Pi, Int * 25 - 150);
    cin.get();
    return 0;
}
```

Ба ғайр аз ин, функцияи мазкур дар дилхоҳ минтақаи саҳифаи тасвир хориҷ намудани натиҷаро фароҳам меорад. Дарозии май-донро бо навиштани як адади бутун дар байни аломати % ва коди қолаб муайян мегардад. Агар хоҳем, ки фазои ҳолӣ бо нол пур карда шавад, пеш аз адади гуфташуда нол менависем. Дар мавриди ададҳои ҳақиқӣ дароз бошад, барои пас аз вергул ҷоп намудани микдори разрядҳои лозимӣ (аниқ), пас аз рамзи % як нуқта ва микдори разрядҳои лозимиро менависем. Масалан, қолаби %8.3f-ададҳо бо ҳашт рамз бо дарозии майдон ва се рақам баъди вергул ҷоп мекунад.

Мисоли 3. *Истифодаи қолаб дар функцияи printf().*

```
#include <iostream>
using namespace std;
int main(){
    double Pi = 3.14159;
    int Int = 25;
    printf("%5.1f % 4d\n", Pi, Int);
    printf("%6.2f % 5d \n", Pi, Int);
    printf("%7.3f % 6d \n", Pi, Int);
    printf("%8.4f % 7d \n", Pi, Int);
    printf("%9.5f % 8d \n", Pi, Int);
    cin.get();
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
3.1 25
3.14 25
3.142 25
3.1416 25
3.141591 25
```

Тавре дида мешавад, дар ин чо якчанд сатр чоп шуд ва дар онҳо дарозии майдон, яъне аз кадом мавқеи саҳифаи тасвир навишта шудани маълумот нишон дода шудааст ва дар адади якум пайдарпай нишон дода шудааст, ки баъд аз вергул чанд разряд гирифта шавад.

Ба ҳамин монанд дарозии майдонро барои дилхоҳ навъи додаҳои сатрӣ ва ададӣ метавон муайян кард. Ин амал барои сатрҳо (риштаҳо) ҳам монанди ададҳо мебошад, яъне агар дарозии сатр аз дарозии майдони пешбинишуда камтар бошад, тарафи чапи он ба микдори лозимии ҷои холӣ (фосила) пур карда мешавад ва дар сурати баръакс, барои пурра намоиш додани сатр ҷои холӣ ҷудо карда мешавад. Ва агар қолабе, ки барои ададҳои ҳақиқӣ ба кор бурдем (яъне баъди вергул микдори разряд ҳои чопшаванда ҳамчун мисоли пешина мушаххас карда шавад) истифода шавад, микдори рамзҳои намоишдодашуда аз ададе, ки пас аз нуқта навишташуда камтар бошад, рамзҳои аз он адад зиёд аз тарафи рост намоиш дода намешаванд ҳарчанд, ки дарозии майдон аз дарозии сатр калонтар бошад.

Мисол. Хориҷ кардани сатр тавассути функсияи `printf()`.

```
#include <iostream>
using namespace std;
int main(){
    char Satr[18] = "Man Barnameasozam";
    printf(" %10s", Satr);
    printf("\n %20s", Satr);
    printf("\n%20.11s", Satr);
    printf("\n%18.14s", Satr);
    printf("\n%.3s\n", Satr);
    cin.get();
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
Man Barnameasozam
      Man Barnameasozam
                Man Barname
                  Man Barnamasoz
                    Man
```

Тавре ки мушоҳида кардем, дар сатри якум ҳарчанд ки дарозии майдон аз дарозии сатри додашуда хурд буд, сатр пурра чоп гардид. Сатри дуюм бошад, бо панҷ фазои холӣ (яъне дарозии майдон аз дарозии сатри додашуда чор рамз бештар буд ва як фазои холигии байни қолабҳо) чоп гардид. Сатри сеюм бошад, ёздаҳ рамзи сатрро ба чоп додаст. Тавре ки қаблан гуфта будем, пас аз нуқта чи қадар қимате омада бошад, ҳамон қадар рамз чоп мегардад, ҳарчанд, ки дарозии майдон аз дарозии сатри додашуда калонтар бошад. Танҳо пеш аз сатр ба микдори зиёдатиаш ҷои холӣ боқӣ мегузорад ва билохира сатри чорум бошад, танҳо се рамзӣ сатрро ба чоп медиҳад, зеро дар қолаб ҳамин қадар нишон дода шудааст.

Функция scanf()

Функция scanf() барои аз клавиатура ба барнома дохил намудани кимати тағйирёбандаҳо истифода бурда мешавад. Аз рӯи нахви истифодабарӣ монанди функцияи printf() буда, баръакси он амал мекунад. Аргументҳои функцияи мазкур метавонанд аз дилхоҳ навъи додаҳо ба монанди додаҳои ададӣ, рамзӣ ва сатрӣ бошанд. Пеш аз ҳар як аргументи он оператори адреси (&) навишта мешавад. Нахви умумии ин функция ба сурати зерин аст:

```
scanf("control string",&arg1&arg2,...,&argn)
```

Тавре ки дида мешавад, нахви умумии ин функция ҳам ба функцияи printf монанд буда, танҳо бо як оператори адрес фарқ мекунад. Ҳангоми додаҳои сатрӣ ро ба барнома ворид намудан пеш аз тағйирёбанлаи сатрӣ ҳоҷати навиштани оператори адрес нест, зеро сатрҳо дар забони C++ ҳамчун массивҳои рамзӣ муаррифӣ карда мешаванд ва номи массив бошад, адреси аввалии онро нишон медиҳад. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    char name[15];
    printf("Input Name ");
    scanf("%s", name);
    printf("Output Name = %s", name);
    cin.get();
    return 0;
}
```

Рамзҳои идоракунандаи функцияи мазкур монанди рамзҳои идоракунандаи функцияи printf() мебошанд

Код	Маънои фармони қолаб(навъи иттилооти чопшаванда)
%c	(character data Item)- Додаҳои навъи рамзӣ
%d	(Signed decimal Integer)-Додаҳои навъи бутуни аломатдор
%i	Додаҳои навъи бутуни аломатдор (Signed decimal integer)
%u	(Unsigned decimal integer)-Додаҳои навъи бутуни беаломат
%f,%e,%g	Додаҳои навъи ҳақиқӣ
%h	Додаҳои навъи бутуни кӯтоҳ (short int)
%s	Додаҳо аз намуди як сатре, ки ҳангоми воридшавӣ тавассути рамз non_white_space оғоз гардида, ба рамзи white_space ба охир мерасад (Ба охири сатр ба таври автомати рамзӣ %0 илова мегардад)
%o	Додаҳо аз намуди бутуни ҳаштӣ
%x,%X	(unsigned hexadecimal integer)- Додаҳо аз навъи адади бутун дар системаи ҳисоби шонздаҳӣ
%p	(display a pointer)-Додаҳо аз навъи ишорагар

Илова бар ин, дар библиотекаи stdio функцияҳои дигари дохилкунӣ ва хоричкунӣ мавҷуданд, ки маъмултаринашонро дида мебароем.

Функция getchar() ва putchar()

Функцияи getchar барои хондани як рамз хизмат карда, дорои аргумент намебошад, балки амали бахшиш (қойгузори)–ро иҷро мекунад, яъне рамзи

воридшударо ба тағйирёбандае, ки дар тарафи чапи он воқеъ аст, бахшида мешавад. Нахви умумии ин функция чунин аст:

```
character variable =getchar() ё Тағйирёбандаи рамзи=getchar()
```

Функцияи putchar барои навиштан ё чоп намудани як рамз дар саҳифаи тасвир хизмат мекунад. Нахви умумии ин функция чунин аст:

```
putchar(Character variable ) ё putchar (Тағйирёбандаи рамзӣ)
```

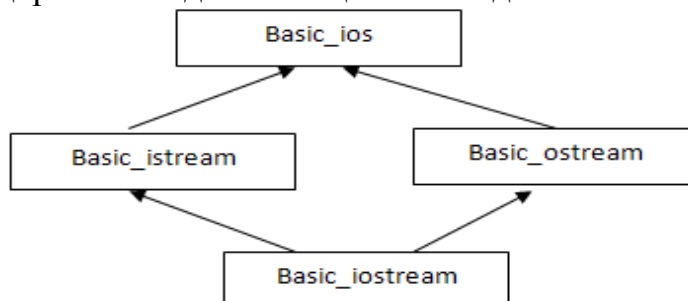
Функцияҳои дохилкунӣ ва хориҷкунӣ дар забони C++

Дар забони C++ дохилкунӣ ва хориҷкунӣ додаҳо ба барнома ҳамчун ҷараёни байтҳо (ҷараён-пайдарпайи байтҳо) сурат мегирад. Ба намуди пешфарз барои дохил нимудани иттилоот ба компютер клавиатура, дисковод, флейшдискҳо, сет, сканер ва ғайра истифода шуда, барои хориҷкунӣ бошад, принтер, дисковод ба ғайра хизмат мекунад.

Дар забони C++ барои пайдарпайии байтҳо ба компютер дохил ва ё аз он хориҷ намудан аз функцияҳои (объектҳои) зиёде истифода мебаранд. Барои аз функцияҳо (объектҳои) истифода бурдан, пеш аз ҳама, библиотекае, ки дар он фармонҳои дохилкунӣ ва хориҷкунӣ мавҷуд аст, пайваст карда мешавад.

```
#include <iostream>
```

Ин китобхона дорои якчанд шаблонҳо мебошад.



Тавре ки аз нақша мушоҳида карда мешавад, шаблонҳои basic_istream ва basic_ostream меросгири рости шаблони базавии basic_ios буда, шаблони iostream бошад, меросбари якчанд синф basic_istream ва basic_ostream мебошад.

Барои дохил намудани маълумотҳо аз объекти cin, ки аз шаблони синфи istream мебошад, истифода бурда мешавад. Сабаби объект ном бурдани cin ва cout дар он аст, ки онҳо объекти классии iostream мебошанд. Калимаи cin шакли тахфифи ду калимаи англисии c-номи забон ё ҳарфи аввали console ва in ду ҳарфи аввали калимаи input-дохилкунӣ буда, маънои аз клавиатура дохил намудани маълумотҳо дорад. Ин чунин амалгарҳои кӯчиш (лағжиш) ба рост (сдвиг вправо) << дар объекти хориҷкунӣ ва лағжиш (кӯчиш) ба чап (сдвиг влево) >> дар объекти хориҷкунӣ истифода бурда мешаванд. Истифодаи амалгарҳои мазкур маъно аз рӯи байтҳо хондан ва навиштани додаҳо дорад.

Функция (Объект)-ҳои хориҷкунӣ

Синфи ostream дар забони C++ имконият медиҳад, ки додаҳо ба таври шаклбандишуда ва бешаклбандишуда ба саҳифаи тасвир ба-роварда шаванд. Синфи мазкур дорои объектҳои хориҷкунӣ зиёде мебошад, ки муҳимтаринашон дар ҷадвали зерин оварда шудаанд.

Номи объект	Маъно ва мафхум
Cout	Барои хориҷкунии додаҳои навъҳои стандартӣ истифода бурда мешавад.
Put	Барои хориҷкунии рамзҳо истифода бурда мешавад.
write	Барои хориҷкунии битҳо (маълумотҳо)
Get	Барои хориҷкунии як рамз истифода бурда мешавад.
getline	Монанди get буда, барои хориҷкунии якчанд рамз истифода бурда мешавад.

Ин объектҳо якҷоя бо объекти яқум, яъне cout истифода бурда мешаванд. Дар оянда ҳар яке аз ин объектҳоро муфассал дида мебароем. Ҳоло бошад, объекти cout-ро мавриди баҳс қарор медиҳем. Тавре ки пештар гуфтем, объекти cout барои аз барнома ба саҳифаи тасвир баровардани маълумотҳои навъҳои стандартии ва инчунин сатрҳо истифода бурда мешавад. Худи объекти мазкур ҳам аз ду калимаи англисии C-номи забон ё ҳарфи аввали калимаи console ва out шакли кӯтоҳшудаи output-хориҷкунии гирифта шудааст. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    cout << "Istifodai obekti cout dar C++";
    cin.get();
    return 0;
}
```

Ҳангоми паҳши тугмачаи F9 дар саҳифаи тасвир пайгоми: Istifodai obekti cout dar C++ чоп карда мешавад.

Тавре ки дидем, пас аз чопи сатри додашуда курсор дар паси матн (сатр) қарор гирифт. Агар хоҳем, ки он дар сатри нав ҷойгир шавад, пас аз сатри навишташуда, пеш аз пӯшидани нохунак рамзи хати қач (слейш-\) ва ҳарфи n-ро мегузорем. Матни барномаи мо намуди зеринро мегирад.

```
#include <iostream>
using namespace std;
int main(){
    cout << "Istifodai obekti cout dar C++\n";
    cin.get();
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
Istifodai obekti cout dar C++
-|
```

Тавре ки мушоҳида гардид, пас аз гузоштани хати қач ва ҳарфи n курсор дар сатри нав қарор гирифт. Ҳарфи n, ки мо дар ин ҷо истифода бурдем, ҳарфи аввали калимаи NewLine-хати нав мебошад. Хати қач бошад, маънои escape (переход) –гузаштан ва ё озод шуданро дорад. Дар забони C++ якчанд рамзҳои (ҳарфҳои) мавҷуданд, ки онҳоро рамзҳои идоракунонда мегӯянд. Ҳоло дар ҷадвали зерин якҷандтои онҳоро меорем.

Рамз	Маъно ва мафҳум
\n	Барои ба сатри нав (newline) кӯчонидани курсор
\t	Барои дар сатри мавҷудбуда сархати горизонталӣ (horizontal tab) гузоштан
\r	Барои ба аввали сатр овардани мавқеи курсор
\a	Садои зангулаи системавӣ (сигнал тревога)
\\	Слеши баракс барои чопи рамзи слеши чаппа
\”	Барои чоп намудани нохунак

Ин рамзҳо қариб дар тамоми компиляторҳои забонҳои оилаи C/C++ якхел истифода бурда мешаванд.

Функцияи put

put функция набуда, балки функция-аъзо (функция-член) мебошад ва он якҷоя бо оператори хориҷкунии cout истифода бурда шуда, имконияти чопи танҳо як рамзро дорад. Бо ибтои дигар, аргументи он танҳо як рамзи ASCII-код мебошад. Масалан, фармони зерин дар саҳифаи тасвир ҳарфи А-ро ба чоп медиҳад.

```
std::cout.put('A');
```

Ин сатри фармони дар саҳифаи тасвир ҳарфи А-ро намош дода, курсор дар пасаш қарор мегирад. Барои курсорро ба сатри навбати овардан аз рамзҳои идоракунонда истифода бурда мешавад.

```
std::cout.put('A').put('\n');
```

Оператори мазкур метавонад, ба сифати аргумент кодҳои ASCII-ро истифода барад. Мисол фармони зерин дар саҳифаи тасвир ҳарфи А-ро намоиш медиҳад, зеро ки ASCII-коди ҳарфи А 65 мебошад.

```
std::cout.put(65).put('\n');
```

Функцияҳои get ва getline

Функцияи get барои дохилкунии як рамз, дар ҷои муайяншуда истифода бурда мешавад. Ин функция ҳангоми дохилкунӣ метавонад дорои аргумент ва ё бе аргумент бошад. Дар сурати истифодабарии аргумент симболи воридкардашуда ба он бахшида мешавад. Ва дар акси ҳол, бояд функцияи мазкур тавассути тағйирёбандаи дигар даъват карда шавад.

Мисол.

```
#include <iostream>
using namespace std;
int main(){
    char chr;
    chr = cin.get();
    cout.put(chr);
    cin.get();
    return 0;
}
```

Тавре ки аз ин мисол бармеояд, функцияи мазкур якҷоя бо оператори дохилкунии cin истифода шуда, қимати воридшударо ба тағйирёбандаи chr бахшид, ҳол он ки мо метавонистем онро ба сурати зерин истифода барем.

```

#include <iostream>
using namespace std;
int main(){
    char chr;
    cin.get(chr);
    cout.put(chr);
    cin.get();
    return 0;
}

```

Тавре ки мушоҳида шуд, дар ин ин чо функсияи `get()` тағйирёбандаи лозимиро ҳамчун аргумент қабул намуда, қимати онро вориди барнома сохт. Агар дар саҳифаи тасвир ҳангоми дохил намудан якчанд рамз нависем, аз он рамзҳо танҳо аввалинашро вориди барнома мекунад халос.

Равиши дуҷуми истифодабарии ин функсия то расидан ба нишонаи интиҳои матн ба аргументи символӣ пайиҳам рамзҳоро мебахшад. Ин равишро баъдтар ҳангоми истифодабарии операторҳои идоракунии дида мебароем.

Равиши сеҷуми истифодаи ин функсия бошад, дорои се параметр мебошад. Массиви рамзӣ (ягон сатр), адади максималии миқдори рамзҳои массив ва рамзи идоракунонда хусусан (`'\n'`). Ин равиши истифодабарӣ рамзҳоро то замоне ки аз миқдори максимали додасуда як воҳид хурданд меҳонад, яъне $(max-1)$ -рамзро меҳонад. **Мисол:**

```

#include <iostream>
using namespace std;
int main(){
    const int Size = 5;
    char Satr[Size];
    cin.get(Satr, Size);
    cout << Satr << "\n";
    cin.get();
    return 0;
}

```

Функсияи `getline ()` бошад, монанди равиши сеҷумӣ буда, чи кадаре ки миқдори максимали бошад, ҳамон қадар рамзро меҳонад. Яъне он рамзи маҳдудкунонда (рамз ограничитель)-ро аз ҷараён нест карда, онро ба массив илова наменамояд.

```

#include <iostream>
#include <conio.h>
using std:: cout;
using std:: cin;
using std:: endl;
int main(){
    const int Size = 5;
    char Satr[Size];
    cin.getline(Satr, Size);
    cout << Satr << "\n";
    cin.get();
    return 0;
}

```

Функсияи мазкур бештар ҳангоми ягон файли маълумотино хондан истифода бурда мешавад, ки дар оянда дар мавзӯи файлҳо муфассал ин матолибро дида мебароем.

Тавре ки мушоҳида кардем, барои истифодабарии ҳар як объекти библиотекаҳои додасуда мо баъд аз муаррифӣ намудани библиотека using std:: номи объект ва ё пеш аз ҳар объект дар дилхоҳ ҷои барнома фармони std:: номи объект-ро менависем. Ин амал метавонад моро дилгир созад ва матни барномаро зиёд гардонад. Ҳоло равишро дида мебароем, ки дар ин ҳолат навиштани ин калимаҳо шарт намегардад. Барои иҷрои ин амал пас аз номи библиотека «.h» менависем, ки дар ин ҷо h ҳарфи аввалаи калимаи header-сарлавҳа мебошад. Тарзи истифодаи ин равиш шакли зеринро дорад:

```
#include <номи китобхона.h>.
```

Мисол #include <iostream.h>

Аз ин ҷо мисоли дар боло овардашударо метавон ба сурати зерин низ навишт.

```
#include <iostream.h>
int main(){
    const int Size = 5;
    char Satr[Size];
    cin.getline(Satr, Size);
    cout << Satr << "\n";
    cin.get();
    return 0;
}
```

Тавре ки мушоҳида кардем, дар ин равиш се сатр коди барномаи ки аз фармонҳои зерин иборат буданд, ихтисор гардид.

```
using std:: cout;
```

```
using std:: cin;
```

```
using std:: endl;
```

Қайд. Дар станарти нави C++ ба ҷойи .h фармони using namespace std навишта мешавад.

Беқолаббанди дохил ва хориҷ намудани маълумотҳо

Бо истифода аз функсияҳои read ва write синфҳои istream ва ostream маълумотҳоро метавонем ба қолиббандӣ дохил ва ё хориҷ кунем. Ҳар яке аз ин функсияҳо метавонад миқдори нишондодасудаи рамзҳоро ворид ва ё хориҷ кунанд. Ҳоло ҳар яке аз ин функсияҳоро диде мебароем.

Функсияи write()-ин функсия якҷои бо оператори cout истифода шуда, дорои ду аргумент массиви рамзӣ ва андозаи он мебошад, яъне миқдори ба саҳифаи тасвир ҷоп намудаи рамзҳо нишон дода мешавад. Агар миқдори рамзҳои массив аз адади нишондодасуда кам бошанд, он гоҳ дар саҳифаи тасвир ба ҷои ҳар як рамз як ҷои холӣ мегузорад. **Мисол:**

```
char chr[] = "ABCDEFGH IJ";
cout.write(chr, 5);
```

Ин мисол 5 рамзи аввалаи массиви chr-ро ба чоп медиҳад. Ҳар як рамз ба адреси аввалаи худ ишора мекунад. Ин мисолро метавонистем ба сурати зерин нависем.

```
cout.write("ABCDEFGHIIJ",5);
```

Функсияи мазкур аз массиви рамзии додасуда метавонад микдори дилхоҳи рамзҳои аввалаашро ба чоп диҳад.

Функсияи read бошад, баръакси функсияи write амал намуда, микдори нишондодасудаи рамзҳоро ба барнома дохил мекунад. Ин функсия ҳам дорои ду аргумент яке рамзи ва яке адади бутуни мусбат мебошад, ки аргументи рамзи адрес ё номи массиви рамзӣ буда, адади бутуни мусбат бошад, микдори рамзҳои лозимро нишон медиҳад. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    char chr[10];
    cin.read(chr, 5);
    cout.write(chr, 5);
    cin.get();
    return 0;
}
```

Функсияи gcount бошад, оид ба микдори рамзҳои охири воридшуда хабар (ахбор) медиҳад. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    char chr[10];
    cin.read(chr, 7);
    cout.write(chr, cin.gcount());
    cin.get();
    return 0;
}
```

Тавре ки дидем, дар ин ҷо микдори рамзи чопшаванда нишон дода нашуда аст, вале он микдоре, ки ҳангоми дохилшавӣ нишон додасуда буд ба чоп бароварда шудааст.

Ҳарфҳои крилӣ дар C++

Дар хати фармони Windows асосан аз кодгузориҳои OEM ва дар ҳуди СО Windows бошад, аз кодгузориҳои ANSI истифода бурда мешавад. Аз ин ҷо дар C++ ҳангоми ба сифати пайғом ё натиҷа хориҷ кардани ҳарфҳои крилӣ, дар вақти иҷроиши барнома ҳарфҳои шикастае пайдо мешаванд, ки хондан ва дарки онҳо ғайриимкон мегардад.

Барои аз ин душвори ҳалос шудан роҳҳои гуногун мавҷуд аст. Масалан, дар Borland C++ Builder одатан аз функсияи CharToOem() истифода бурда мешавад. Дар забони Dev C++ бошад, аз барномаҳои gccrus.exe ва g++rus.exe.

Функсияи CharToOem() функсияи мантиқӣ буда, барои сатри додасударо бо стандарти ANSI мувофиқ ё табдил додан истифода бурда мешавад. Наҳви умумии функсияи мазкур намуди зеринро дорад:

bool CharToOem(STR Src, STR Dst)

дар ин ҷо Src сатри додашуда ва Dst ҳофиза (буфер) барои нигоҳ доштани сатри табдилдодашуда мебошад. Барои аз функсияи мазкур истифода кардан бояд китобхонаи windows.h пайваст карда шавад.

Ҳангоми истифодаи функсияи мазкур маслиҳат он аст, ки функсияе муаррифӣ карда шуда, дар ҳолати лозими аз он истифода бурда шавад. Мисол, дохил ва хориҷ кардани ҳарфҳои крилӣ.

```
#include <iostream>
#include <windows.h>
using namespace std;
char* Krill(char* s) {
    static char buf[500];
    CharToOem(s, buf);
    return buf;
}
int main() {
    cout << Krill("Номатонро дохил кунед:\t");
    char Name[100];
    cin >> Name;
    cout << Krill("Салом\t");
    cout << Name << "!" << endl;
    cin.get();
    return 0;
}
```

Дар ин ҷо аввал бо ними Krill функсияе навишта шуда, сипас он дар функсияи асосӣ истифода шудааст. Натиҷаи барнома намуди зеринро дорад:

Номатонро дохил кунед: Саид

Салом Саид!

Дар барномаҳои корбурди консолӣ ҳарфҳои русӣ ҳамчун стандарти рамзгузории CP-866 намоиш дода мешаванд. Дар муқоваи Dev-C++ бошад, ҳамчун кодгузории CP-1251 истифода мешаванд. Ин маънои онро дорад, ки ҳангоми истифодаи ҳарфҳои крилӣ дар саҳифаи тасвир ҳарфҳои шикастае пайдо мешаванд, имконияти хондани онҳо вучуд надорад. Барои аз ин душвори раҳо шудан аз барномаҳои (файлҳои) gccrus.exe ва g++rus.exe истифода бурда мешавад.

Алгоритми истифодаи ин барномаҳо дар Dev-C++ шакли зеринро дорад:

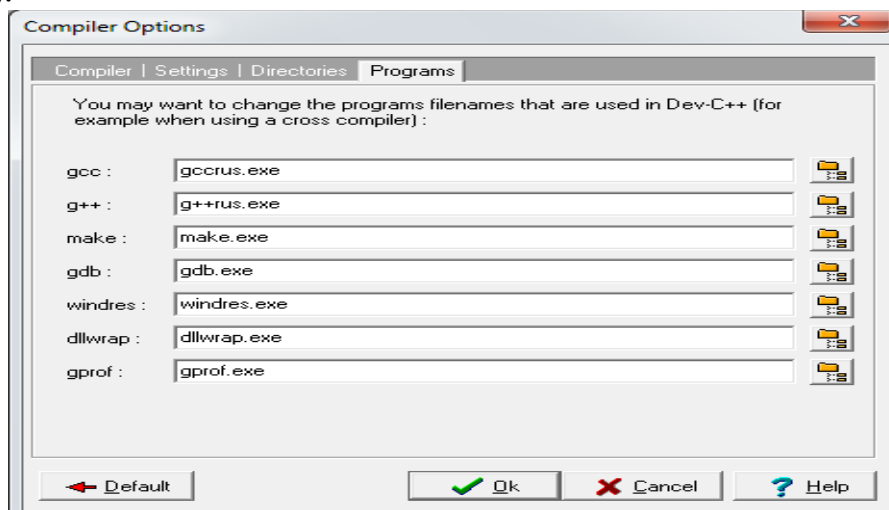
1) Файли архиве, ки дар дохили он барномаҳои gccrus.exe ва g++rus.exe ҷойгиранд, аз интернет берун меоварем. Ҳаҷми архив тақрибан 193 Кб аст.

2) Барномаҳои gccrus.exe ва g++rus.exe –ро дар суроғи зерин нусхабардорӣ мекунем: “C:\Dev-Cpp\bin” (яъне дар каталоги bin, ки зеркаталоги каталоге ки дар он ҷо Dev-C++ насб шудааст.)

3) Dev-C++-ро ба кор омода сохта, аз менюи Tools (Сервис) фармони Compiler Options (Параметры компилятора)-ро иҷро мекунем.

4) Дар варақии Programs номи ду барномаи авваларо ба таври зерин тағйир медиҳем, яъне тугмачаи охири сатри gcc ва g++ -ро пахш карда, аз онҷо

мувофиқан файлҳои gccrus.exe ва g++rus.exe –ро аз каталоги bin мекушоем (расми зерин).



5) Баъд аз иҷрои ин амалиёт бе душворӣ метавонем аз ҳарфҳои крилӣ истифода барем.

Мисол. *Истифодаи ҳарфҳои крилӣ дар Dev c++*

```
#include <cstdlib>
#include <iostream>
using namespace std;
int main() {
    cout << "Номатонро дохил кунед:\t";
    char Name[100];
    cin >> Name;
    cout << "Салом\t";
    cout << Name << "!" << endl;
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

Номатонро дохил кунед: Саид

Салом Саид!

Саволҳо барои мустаҳкамкунӣ

- 1) Функсияҳои дохилкунӣ ва хориҷкунӣ дар забони C кадомҳоянд?
- 2) Объякти cin аз функсияи scanf() чӣ фарқ дорад?
- 3) Насби библиотека магар зарур аст?
- 4) Оё мумкин аст дар як барнома ҳангоми пайваст намудани ягон библиотека ҳам .h ва ҳам using namespace std истифода бурда шавад?
- 5) Дар функсияҳои дохилкунӣ хориҷкунӣ кадом синфҳо дар забони C++ мавҷуданд?
- 6) Синфи ifstream чӣ гуна синф аст?
- 7) Библиотекаҳои C++ дар қучо нигоҳ дошта мешаванд?
- 8) Оператори std::cout чӣ амалро иҷро мекунад?
- 9) Чи тавр ҳарфҳои крилли дар саҳифаи тасвир хориҷ карда мешаванд?
- 10) Чи тавр ҳарфҳои крилӣ дар C++ истифода бурда мешаванд?

Намунаи ҳалли масъалаҳо

1) Тарафҳои росткунҷа а ва b дода шудааст. Барномае нависед, ки периметри росткунҷа-ро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int main() {
    float a, b, P;
    cout << "Adadhoi a va b - ro dohil kuned!\n";
    cin >> a >> b;
    P = a*b;
    cout << "Perimetri rostkunja ba " << P << " barobar ast.\n";
    return 0;
}
```

2) Дарозии тарафҳои секунҷа a,b,c дода шудаанд. Барномае нависед, ки бо истифода аз формулаи Герон масоҳати онро ҳисоб кунад.

Қайд: *Формулаи Герон намуди зеринро дорад,*

$$S = \sqrt{r(r-a)(r-b)(r-c)}$$

ки дар ин ҷо r -нимпериметр мебошад. Ва аз геометрия маълум аст, ки $P=a+b+c$ ва $r = \frac{P}{2} = \frac{a+b+c}{2}$

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    float a, b, c, r, S;
    cout << "Adadhoi a b va c - ro dohil kuned!\n";
    cin >> a >> b >> c;
    r = (a + b + c) / 2;
    S = sqrt(r*(r-a)*(r-b)*(r-c));
    cout << " S = " << S;
    return 0;
}
```

3) Ададҳои бутуни b ва p дода шудаанд. Барномае нависед, ки p фоизи адади b-ро муайян кунад.

Қайд. *«Барои p фоизи адади b-ро ёфтан p ва b-ро бо ҳам зарб намуда, натиҷаро ба 100 тақсим кардан лозим аст.»*

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int p, b;
    float s;
    cin >> p >> b;
    s = p*(float)b / 100;
    cout << p << " Foizi adadi " << b << " ba " << s << " barobar
ast.\n";
    return 0;
}
```

4) Адади f дар қолаби хақиқӣ дода шудааст. Барномае нависед, ки он-ро ба қолаби пулӣ гардонад.

```
#include <stdio.h>
#include <iostream>
using namespace std;
int main() {
    int r, k;
    float f;
    printf("adadi hakiki - ro dohil kuned->");
    scanf("%f", &f);
    r = (int)f;
    k = f * 100 - r * 100;
    printf("%6.2f in %i somoni va %i diram meboshad\n", f, r, k);
    cin.get();
    return 0;
}
```

5) Адади серақмае дода шудааст. Барномае нависед, ки суммаи рақамҳои адад ва ҳосили зарби рақамҳои ададро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int main() {
    int a, b, max, c;
    cout << "Adadi seraqamai a - ro dohil kuned!\n";
    cin >> a;
    b = (a / 100) + (a % 100 % 10) + (a % 100 / 10);
    c = (a / 100)*(a % 100 % 10)*(a % 100 / 10);
    cout << "Summai raqamho = " << b << ": Zarbi raqamho = " << c <<
"\n";
    return 0;
}
```

6) Функсияи $f(x)=\sin x+\cos x$ дода шудааст. Барномае нависед, ки бо дода шудани қиматҳои a ва b қимати ифодаи зеринро ҳисоб кунад.

$$Z=f(f(a+b))$$

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    float a, b, Y, Z;
    cout << "Qimarhoi a va b - ro dohil kuned!\n";
    cin >> a >> b;
    Y = sin(a + b) + cos(a + b);
    Z = sin(Y) + cos(Y);
    cout << "Z = " << Z << "\n";
    return 0;
}
```

8) Солшумории ҳичрӣ аз соли 622 аз рӯзи ҳичрати Пайғомбари Ислон (Муҳаммад (с)) оғоз гардидааст. Барномае нависед, ки дилхоҳ соли ҳичриро ба мелодӣ гардонад.

Қайд. Метавонем барои баргардонидани ин солишуморӣ аз формулаҳои зерин истифода барем. $M = X - \frac{X}{33} + 62$ $X = M - 622 + \frac{M-622}{32}$

```
#include <iostream>
using namespace std;
int main() {
    int H, M;
    cout << "Soli hijriro dohil kuned!\n";
    cin >> H;
    M = (int)(H - H / 33 + 622);
    cout << "M = " << M << "\n" << "H = " << H << "\n";
    return 0;
}
```

9) Барномае нависед, ки андоза ё тӯли ҳамаи навъҳои додаҳоро ба ҳисоби байт муайян кунад.

Қайд. Ин барнома ро бо ёрии оператори ҳофиза `sizeof()` иҷро мекунем.

```
#include <iostream>
using namespace std;
int main() {
    long double S, r, h;
    cout << "Darozii tiphoi dodaho!\n\n";
    cout << "RT\tNamei tip    andozai tip\n";
    cout << "1\tchar\t\t" << sizeof(char) << " byte\n";
    cout << "2\tshort\t\t" << sizeof(short) << " byte\n";
    cout << "3\tint\t\t" << sizeof(int) << " byte\n";
    cout << "4\tlong\t\t" << sizeof(long) << " byte\n";
    cout << "5\tfloat\t\t" << sizeof(float) << " byte\n";
    cout << "6\tdouble\t\t" << sizeof(double) << " byte\n";
    cout << "7\tlong double\t" << sizeof(long double) << " byte\n";
    return 0;
}
```

10) Шахсе дар бонк бо фоизи солонаи $P\%$ S миқдор пул гузошт. Барномае нависед, то муайян кунед, ки пули гузоштаи он дар муддати вақти t чӣ қадар зиёд мешавад ва чӣ қадар даромад меоварад?

```
#include <iostream>
using namespace std;
int main() {
    float summ; //Пули гузошташуда
    int muhl; //Муҳлати гузоштан
    float P; //Фоизи зиёдшави
    float dar; // Даромад аз пули гузошташуда
    cout << "Miqdori puli guzoshtashuda\n";
    cin >> summ;
    cout << "Muhlati puli guzoshtashuda\n";
    cin >> muhl;
    cout << "Foizi ziyodshavii pul\n";
    cin >> P;
    dar = summ*P / 365 / 100 * muhl; //Ҳисоб намудани даромад
    summ = summ + dar; //Даромад бо пули гузошташуда якҷоя
```

```

cout << "daromad = " << dar << " somoni\n";
cout << "Puli ziyodshuda = " << summ << " somoni\n";
return 0;
}

```

Қайд. Ҳамаи барномаҳои дар дастур овардашударо метавон дар дигар компиляторҳои C++ бо таҳрири кучаке навишт. Масалан, барномаи мисоли 1-ро агар дар компилятори Microsoft Visual C++ нависем шакли зеринро мегирад.

```

#include "stdafx.h"
#include <iostream>
using namespace std;
int _tmain(int argc, _TCHAR* argv[]){
    float a, b, P;
    cout << "Adadhoi a va b-ro dohil kuned!\n";
    cin >> a >> b;
    P = a*b;
    cout << "Perimetri rostkunja ba " << P << " barobar ast.\n";
    cin.get();
    return 0;
}

```

Тавре ки дида мешавад, дар ин ҷо ибтидо китобхонаи мавҷуд буда, функсияи асосӣ шакли `int _tmain(int argc, _TCHAR* argv[])`-ро дорад. Илова бар ин ба ҷойи функсияи `getch()` аз функсия-аъзои `cin.get()` истифода карда шудааст.

Дар саросари ин дастури таълимӣ ҳарчанд, ки барои фаъол гардонидани барнома аз “.h” ва барои нигоҳ доштани равшани вуруд-хурӯҷӣ аз `getch()` истифода бурда шудааст, вале маслиҳат дода мешавад, ки равиши намунаи мисоли дар қайд овардашуда истифода карда шавад.

Барои иҷро намудани барнома дар компилятори Microsoft Visual C++ тугмачаи F5 пахш карда мешавад.

Масъалаҳо барои кори мустақилона Масъалаҳои гурӯҳи Begin

1. Ду адади натуралӣ дода шудааст. Барномае нависед, ки қимати миёнаи арифметикӣ ва геометрии онҳоро ҳисоб кунад.
2. Адади серақама дода шудааст. Барномае нависед, ки ҷойи разрядҳои садӣ ва даҳии онро иваз кунад.
3. Ададҳои бутуни h ва m ($0 < h \leq 12, 0 \leq m \leq 59$) дода шудааст, ки дар ин ҷо лаҳзаи вақтро h бо соат ва m бо дақиқа ифода мекунад. Вақти кӯтоҳтарине (бо дақиқа), ки то бо ҳам вохӯрдани акрабакҳо мегузарад, муайян карда шавад.
4. Тарафҳои росткунҷа a ва b дода шудааст. Барномае нависед, ки периметри онро ҳисоб кунад.

Қайд. Периметри росткунҷа бо ёрии формулаи $p = 2 * (a + b)$ ҳисоб карда мешавад.

5. Ду адади бутун пайихам ворид карда мешаванд. Барномае нависед, ки дар ҳолати тоқ будани онҳо қимати 1, дар дигар ҳолатҳо қимати аз як фаркунанда хориҷ кунад.

Қайд. *Аз оператори шартӣ ва даврӣ истифода бурдан мумкин нест.*

6. Адади чоррақама дода шудааст. Ададе, ки хангоми ҷойивазкунии рақами якум бар сеюм ва дуум бар чорум пайдо мешавад, ёфта шавад.
7. Адади серақама дода шудааст. Барномае нависед, ки қимати миёнаи арифметикӣ ва геометрии рақамҳои онро ҳисоб кунад.
8. Аз оғози шабонарӯз акрабаки соат ба y градус ($0 \leq y \leq 360$, y - адади бутун) давр зад. Муайян карда шавад, ки аз оғози шабонарӯз чанд дақиқаи пурра гузаштааст.
9. Адади дурақама дода шудааст. Барномае нависед, ки суммаи квадрати рақамҳои онро ҳисоб кунад.
10. Адади бутуни b ва p дода шудаанд. Барномае нависед, ки p фоизи адади b -ро муайян кунад.

Қайд. *Барои p фоизи адади b -ро ёфтан p ва b -ро бо ҳам зарб намуда, натиҷаро ба 100 тақсим кардан лозим аст.*

11. Адади f дар қолаби ҳақиқӣ дода шудааст. Барномае нависед, ки онро ба қолаби пулӣ гардонад.
12. Функцияи $f(x) = \sin x + \cos x$ дода шудааст. Барномае нависед, ки бо дода шудани қиматҳои a ва b қимати ифодаи $Z = f(f(a+b))$ -ро ҳисоб кунад.
13. Солшумории ҳичрӣ аз соли 622 - и мелодӣ оғоз гардидааст. Барномае нависед, ки дилхоҳ соли ҳичрӣро ба мелодӣ гардонад.

Қайд. *Барои соли ҳичрӣро ба мелодӣ гардонидан, аз формулаи $M = X - \frac{X}{33} + 62$ ва соли мелодиро ба ҳичрӣ гардонидан, аз формулаи $X = M - 622 + \frac{M-622}{32}$ истифода карда мешавад.*

14. Шахсе дар бонк S - сомони бо фоизи солони P гузошт. Муайян кунед, ки пули гузоштаи он дар муддати вақти t чӣ қадар зиёд мешавад ва чӣ қадар даромад меоварад?
15. Суммаи n аъзои прогрессияи арифметикиро бо дода шудани аъзои нахустин ва фарқи он ҳисоб кунед.
16. Координатаҳои се қуллаи секунҷа дода шудааст. $(x_1, y_1), (x_2, y_2), (x_3, y_3)$. Масоҳат ва периметри онро ёбед.
17. Баландии секунҷаро бо дода шудани тарафҳои он a, b, c ёбед.
18. Нависандаи амрикоӣ Рэя Бредбери романи бо номи "450 градус бо Фаренгейт" навиштааст. Ин ҳароратро бо селсия гардонед.
19. Суммаи n аъзои прогрессияи геометрӣро бо дода шудани аъзои нахустин ва маҳраҷи он ҳисоб кунед.
20. Ҳалли системаи муодилаи

$$\begin{cases} A_1 \cdot x + B_1 \cdot y = C_1, \\ A_2 \cdot x + B_2 \cdot y = C_2, \end{cases}$$

-ро ҳангоми мавҷуд будани реша, бо дода шудани коэффитсиентҳо $A_1, B_1, C_1, A_2, B_2, C_2$ бо истифода аз формулаҳои $x = (C_1 \cdot B_2 - C_2 \cdot B_1)/D$, $y = (A_1 \cdot C_2 - A_2 \cdot C_1)/D$ (дар ин ҷо $D = A_1 \cdot B_2 - A_2 \cdot B_1$ аст) ҳисоб кунед.

Масъалаҳои гурӯҳи Integer

- 1) Адади серақама дода шудааст. Ададе ёфта шавад, ки хангоми ҷойивазкунии рақамҳои якуму дууми он ҳосил мешавад (123 натиҷа: 213).

- 2) Адади чоррақама дода шудааст. Ҳосили зарб ва ҳосили чамъи квадрати рақамҳои он ёфта шавад.
- 3) Адади чоррақама дода шудааст. Ададе, ки ҳангоми аз рост ба чап хондани ин адад пайдо мешавад ёфта шавад. (1234 натиҷа: 4321)
- 4) Адади чоррақама дода шудааст. Ададе, ки ҳангоми ҷойивазкунии рақами якум бар дуҷум ва сеҷум бар чорум пайдо мешавад ёфта шавад. Масалан аз адади 1234 адади 2143 пайдо мешавад.
- 5) Аз адади сарақамаи x рақами якуми он партофта шуд. Адади боқимонда ба 10 зарб карда, пас аз он ба рақами сеҷуми партофташуда чамъ карда шуд. Дар натиҷа адади 564 ҳосил гардид. Адади x -ро ёбед.
- 6) Аз адади сарақамаи x рақами дуҷуми он партофта шуд. Ҳангоми ба адади ҳосилшудаи дурақама аз тарафи рост рақами дуҷуми x -ро ҳамроҳ намуданд адади 456 ҳосил шуд. Адади x -ро ёбед.
- 7) Аз адади сарақамаи x рақами охири он партофта шуд. Пас аз он амали ҷойивазкунии рақамҳо дар адади дурақамаи ҳосилшуда иҷро гашт. Ба ин адад аз тарафи чап рақами охири партофташудаи адади x ҳамроҳ карда шуд. Дар натиҷа адади n ҳосил гардид. Ҳангоми дохил намудани қимати n адади x ёфта шавад ($1 \leq n \leq 999$).
10. Рақами соли муайян дода шудааст (адади бутуни мусбат). Рақами мувофиқи асри он муайян карда шавад. Ба назар гирифта шавад, ки оғози асри 20-ум соли 1901 аст.
11. Аз оғози шабонарӯз акрабаки соат ба y градус ($0 \leq y \leq 360$, y - адади бутун) давр зад. Муайян карда шавад, ки аз оғози шабонарӯз чанд дақиқаи пурра гузашт.
12. Адади дурақама дода шудааст. Сумма ва ҳосили зарби рақамҳои онро ёбед.
13. Адади сарақама дода шудааст. Як амали тақсими бақиянокро истифода бурда, рақами сadiи онро хориҷ кунед.
14. Адади n ($n > 9999$) дода шудааст. Як амали тақсими бақиянокро истифода бурда, рақами ҳазории онро хориҷ кунед.
15. Адади сарақама дода шудааст. Аввал рақами воҳид (охирон), сипас, рақами даҳии онро хориҷ кунед.
16. Адади сарақама дода шудааст. Сумма ва ҳосили зарби рақамҳои онро ҳисоб кунед.
17. Адади сарақама дода шудааст. Рақами нахустини онро хат зада, дар охири он нависед.
18. Адади сарақама дода шудааст. Рақами охири онро хат зада, дар аввал нависед.
19. Рӯзҳои ҳафта чунин рақамгузорӣ шудаанд: 1-душанбе, 2-сешанбе, 3-чоршанбе, 4-панҷшанбе, 5-ҷумъа, 6-шанбе, 7-якшанбе. Адади бутуни K , ки дар фосилаи 1-365 меҳобад дода шудааст. Рақами ҳафтаи рузи K -уми сол ёфта шавад, агар маълум бошад, ки дар ин сол 1-уми январ рӯзи ... буд.
 - a) якшанбе
 - b) душанбе
 - c) сешанбе
 - d) чоршанбе

- e) панҷшанбе
 - f) чумъа
 - g) шанбе
20. Рӯзҳои ҳафта чунин рақамгузорӣ шудаанд: 1- душанбе, 2- сешанбе, 3- чоршанбе, 4- панҷшанбе, 5- чумъа, 6- шанбе, 7- якшанбе. Адади бутуни K , ки дар фосилаи 1-365 ва адади бутуни N , ки дар фосилаи 1-7 меҳобад дода шудааст. Рақами ҳафтаи рузи K -уми сол ёфта шавад, агар маълум бошад, ки дар ин сол 1-уми январ рӯзи ҳафтаи N -ум буд.

Масъалаҳои гурӯҳи Boolean

1. Адади серақама дода шудааст. Муайян кунед, ки оё адади серақама аз рост ба чап ва аз чап ба рост якхел хонда мешавад.
2. Ададҳои мусбати бутуни A ва B дода шудаанд. Дурустии мулоҳизаи «Ақалан яке аз ададҳои A ва B тоқ нестанд» санчида шавад.
3. Адади бутуни a дода шудааст. Дурустии мулоҳизаи зеринро санҷед. «Адади a ... мебошад»
 - a) «мусбат»
 - b) «манфӣ»
 - c) «тоқ»
 - d) «чуфт»
 - e) «дурақамаи чуфт»
 - f) «дурақамаи тоқ»
4. Адади серақамаи n дода шудааст. Дурустии мулоҳизаи зеринро санҷед.
 - a) «Суммаи рақамҳои адади n адади чуфт мебошад»
 - b) «Қимати миёнаи арифметикии рақамҳои адади n адади тоқ мебошад»
 - c) «Адади n аз рост ва аз чап якхел хонда мешавад (Палиндром).»
 - d) «Рақамҳои адади n бо тартиби афзуншавӣ ҷойгир мебошанд»
 - e) «Рақамҳои адади n бо тартиби камшавӣ ҷойгир мебошанд»
5. Се адади натуралии a , b ва c дода шудааст. Дурустии мулоҳизаи зеринро санҷед. «Секунҷаи тарафҳояш ба a , b ва c баробар мавҷуд аст.»
6. Координатаҳои ягон каталчаи тахтаи (майдони) шохмот x ва y (ададҳои бутуни аз 1 то 8) дода шудаанд. Бо назардошти он ки каталчаи поёнии тарафи ростии тахтаи шохмот (1,1) сиёҳ аст, дурустии мулоҳизаи зеринро санҷед. «Каталчаи (майдони) додасуда ранги сафед дорад.»
7. Координатаҳои ду каталчаи тахтаи (майдони) шохмот (x_1, y_1) (x_2, y_2) (ададҳои бутуни аз 1 то 8) дода шудаанд. Бо назардошти он ки каталчаи поёнии тарафи ростии тахтаи шохмот (1,1) сиёҳ аст, дурустии мулоҳизаи зеринро санҷед. «Майдонҳои додасуда ранги якхела доранд.»
8. Ададҳои бутуни $a(a \neq 0)$, b ва c дода шудаанд. Дискриминант $(d = b^2 - 4ac)$ -ро ба назар гирифта, дурустии мулоҳизаи «муодилаи квадратии $ax^2 + bx + c = 0$ решаҳои ҳақиқӣ дорад» санчида шавад.
9. Ду адади бутуни A ва B дода шудаанд. Дурустии мулоҳизаи зеринро санҷед: «Нобаробариҳои $A > 2$ ва $B \geq 3$ ҷой доранд.»
10. Ду адади бутуни A ва B дода шудаанд. Дурустии мулоҳизаи зеринро санҷед: «Нобаробариҳои $A > 0$ ё $B \leq -2$ ҷой доранд.»

11. Се адади бутуни A , B ва C дода шудаанд. Дурустии мулоҳизаи зеринро санҷед: «Нобаробарии $A < B < C$ чой дорад».
12. Ду адади бутуни x ва y , ки координатаҳои нуқтаро дар ҳамворӣ ифода мекунад, дода шудаанд. Дурустии мулоҳизаи зеринро санҷед.
- «Нуқта бо координатаҳои x ва y дар чоряки якуми ҳамвории декартӣ мехобад.»
 - «Нуқта бо координатаҳои x ва y дар чоряки дууми ҳамвории декартӣ мехобад.»
 - «Нуқта бо координатаҳои x ва y дар чоряки сеюми ҳамвории декартӣ мехобад.»
 - «Нуқта бо координатаҳои x ва y дар чоряки чоруми ҳамвории декартӣ мехобад.»
 - «Нуқта бо координатаҳои x ва y дар чоряки якуми ё дууми ҳамвории декартӣ мехобад.»
 - «Нуқта бо координатаҳои x ва y дар чоряки якуми ё сеюми ҳамвории декартӣ мехобад.»
 - «Нуқта бо координатаҳои x ва y дар чоряки дуум ё чоруми ҳамвории декартӣ мехобад.»
13. Координатаҳои ду катакҷаи тахтаи (майдони) шохмот (x_1, y_1), (x_2, y_2) (ададҳои бутуни аз 1 то 8) дода шудаанд. Дурустии мулоҳизаи зеринро санҷед: «... бо як ход метавонад аз катакҷаи (x_1, y_1) ба катакҷаи (x_2, y_2) гузарад.»
- «шоҳ»
 - «фил»
 - «асп»
 - «пиёдагард»
 - «фарзин»
 - «руҳ»
14. Майдони тахтаи шохмот бо ҷуфти ададҳои натуралӣ (дар диапазони 1-8 хобанда) муайян карда мешавад. Адади якум рақами амуди ро (ҳангоми аз поён ба боло шуморидан) ва адади дуум рақами уфуқиро (ҳангоми аз чап ба рост шуморидан) ифода мекунад. Ададҳои натуралии a, b, c, d дода шудаанд.
- дар майдони (a, b) фил истодааст. Муайян карда шавад, ки фил ба майдони (c, d) таҳдид карда метавонад ё не?
 - дар майдони (a, b) шоҳ истодааст. Муайян карда шавад, ки оё шоҳ бо як равиш (ҳаракат) ба майдони (c, d) гузашта метавонад ё не?
 - дар майдони (a, b) руҳ истодааст. Муайян карда шавад, ки оё руҳ бо як равиш (ҳаракат) ба майдони (c, d) гузашта метавонад ё не?
 - дар майдони (a, b) пиёдагард истодааст. Муайян карда шавад, ки оё пиёдагард бо як равиш (ҳаракат) ба майдони (c, d) гузашта метавонад ё не?

Боби 4. Сохтори операторҳои идоракунии

Муқаддима

Дар ҳар забони барномасозӣ операторҳои идоракунии мавқеи махсусро ишғол мекунанд. Ин операторҳо ба компилятор (ва ё барномасоз) имконият медиҳанд, ки як қитъаи барнома аз рӯи шарти гузошташуда як ё якчанд маротиба иҷро намояд. Тавре ки дар мавзӯи операторҳо гуфта будем, операторҳои мураккаб операторҳои буданд, ки бо ин операторҳо масъалаҳои ҳал мешаванд, ки бо операторҳои идоракунии дохил мешаванд. Шумо бо операторҳои даврии `for...next`, `while` ва операторҳои шартии `if..then` (инчунин `if...then...else`) дар забонҳои Basic, Паскал ва ё дигар забонҳо шояд шинос бошед. Забони C++ нисбат ба забонҳои номбурда ин операторҳоро ба таври васеътар истифода мекунанд. Барои бо ин операторҳо сарукар гирифтани, бояд ду мафҳуми `true` ва `false` –ро дар ёд гирифт, зеро ҳар оператори идоракунии рӯи шарти гузошташуда амал мекунад ва то замоне ки шарт дуруст аст иҷро мегардад. Тавре ки дар мавзӯҳои гузашта қайд кардем, дар забони C++ ҳар гуна адади ғайринулии ҳамчун қимати ҳақиқат фаҳмида шуда, қимати ноҳақиқат бошад, адади нол аст. Операторҳои идоракунии дар забони C++ инҳо мебошанд:

- 1) Операторҳои шартӣ ё сохторҳои тасмимгирӣ (`decision making constructs`): `switch`, `if`
- 2) Операторҳои такрорӣ (`loop iteration constructs`): `do/while`, `while`, `for`
- 3) Операторҳои `goto`, `continue`, `break`
- 4) Операторҳои `return`-барои баргардонидани қимат аз зербарнома ба барномаи асли.

Операторҳои шартӣ

Яке аз намудҳои алгоритми шартӣ мебошад, ки он дар забони C++ истифодаи васеъ дорад. Алгоритми мазкур дар забони C++ бо ёрии ду оператор `switch` ва `if` иҷро карда мешавад. Ба монанди ин навъи алгоритм, операторҳои шартӣ операторҳои мебошанд, ҳангоми ҷой доштани шарт баъзе қисматҳои барнома иҷро мешаванду баъзеашон не.

- 1) Оператори `if`-оператори шартӣ буда, бо ду тарз истифода бурда мешавад: `if` ва `else if`. Навъи якум дар ҳолате истифода бурда мешавад, ки барнома аз як шарт иборат бошад. Навъи дуюм бошад, барои ҳалли масъалаҳои, ки дорои якчанд шарт мебошанд, истифода бурда мешаванд. Тарзи умумии истифодабарии навъи якум яъне `if` ба сурати зерин аст:

<code>if (condition) statement;</code>	ё	<code>if (ифодаи мантиқӣ) оператор;</code>
--	---	---

Дар ин оператор ҳангоми дуруст будани шарти гузошташуда, операторҳои баъди шарт иҷро мегарданд. Агар пас аз оператори `if (condition)` симболи нуқта вергул гузорем, дар ин сурат шарт шартӣ маҳдуд номида шуда, иҷро намегардад. Дар ҳолати акс, операторҳои баъди шарт иҷро мегарданд. Агар ҳангоми иҷрои шарт якчанд операторҳо вобаста ба шартро иҷро карданӣ шавем, он гоҳ операторҳои мазкур дар дохили қавси системавӣ навишта мешаванд, яъне:

if (condition) { statement; }	ё	if (ифодаи мантиқӣ) { оператор; }
--	---	---

Тавре ки мушоҳида кардем, агар пас аз шарт зарурати иҷроиши якчанд ифода (дастур) пеш ояд, он гоҳ, дастурҳои мавриди назар дар дохили қавси фигуравӣ навишта мешаванд. Ин оператор ҳам монанди оператори пешина ҳангоми ҳақ будани шarti гузошташуда операторҳои дохили қавси системавӣ иҷро мешаванд ва дар сурати нодуруст будани шарт операторҳои берун аз қавси системавӣ иҷро карда мешаванд.

Мисоли 1. Истифодаи оператори шартӣ.

```
#include <iostream>
using namespace std;
int main(){
    int a, b, S = 0, Z = 0;
    cout << "Adadhoi a va b - ro dohil kuned!\n";
    cin >> a >> b;
    if (a > b)
    {
        Z = 2 * a;
    }
    S = 3 * b;
    cout << "S = " << S << "   Z = " << Z << "\n";
    return 0;
}
```

Қайд: Дар ин барнома аввал қиматҳои a ва b вориди барнома карда мешаванд ва шarti $a > b$ санҷида мешавад. Дар сурати нодуруст (*false*) будани ин шарт, ифодаи $S = 3 * b$ иҷро гардида, қимати $S = 3 * b$ ва $Z = 0$ ба чоп дода мешаванд. Дар сурати дуруст (*true*) будани шarti гузошташуда аввал ифодаи $Z = 2 * a$ ва баъд ифодаи $S = 3 * b$ ҳисоб шуда, ба чоп дода мешаванд.

Adadhoi a va b-ro dohil kuned! 1 5 S=15 Z=0	Adadhoi a va b-ro dohil kuned! 5 1 S=3 Z=10
--	--

а) Дар ҳолати $a > b$ будан б) Дар ҳолати $a < b$ будан

Дар ин мисол тавре ки дидем, оператори шартӣ аз рӯи шarti гузошташуда амал мекардааст. Аз расми а) дида мешавад, ки ҳангоми a аз b калон будан ифодаи $Z = 2 * a$ сарфи назар шуда, танҳо қимати ифодаи $S = 3 * b$ ҳисоб карда шудааст. Агар барномаи мо аз якчанд шартҳои бо ҳам вобаста иборат бошад, дар ин сурат ин оператор ба кор намеояд. Масалан, бигзор ҳисоб намудани решаҳои муодилаи квадратӣ талаб карда шавад. Аз нуқтаи назари математика чи тавре. Ки медонед муодилаи квадратӣ дорои се шарт ё ҳолат мебошад.

- 1) Ҳолати $D < 0$ будан «муодила ҳақиқӣ ҳақиқӣ надорад».
- 2) Ҳолати $D > 0$ будан, муодила дорои ду решаи ҳақиқӣ аст, яъне x_1, x_2 .
- 3) Ҳолати $D = 0$ будан, решаҳои муодила баробаранд, яъне $x_1 = x_2$.

Барои ҳалли ин гуна масъалаҳо ва масъалаҳои, ки дорои якчанд шарт мебошанд, аз намуди дуҷуми оператори `if`, ки `if_ else` ном дорад, истифода мекорем. Тарзи истифодабарии ин оператор чунин аст:

<pre>if (condition) { statements1; } else { Statements2 } Next statements</pre>	<pre>if (ифодаи мантиқӣ) { Маҷмӯи операторҳои1; } else { Маҷмӯи операторҳои2; } Маҷмӯи операторҳои дигар</pre>
---	---

Дар ин ҷо ҳам `condition`-ифода (ифода)-и мантиқӣ буда, ҳангоми дуруст (`true`) будани он маҷмӯи операторҳои дохили қавси якум иҷро мешаванд ва идоракунӣ ба сатри маҷмӯи операторҳои дигар (`Next statements`) мегузарад. Ҳангоми нодуруст (`false`) будани он ифодаи дохили қавси системавии 2 яъне маҷмӯи операторҳои 2 (`Statements2`) иҷро мегарданд.

Мисоли 2. Ду адади бутун дода шудааст, хурдтарини онҳоро ёбед.

```
#include <iostream>
using namespace std;
int main(){
    int a, b, min;
    cout << "Adadhoi a va b - ro dohil kuned!\n";
    cin >> a >> b;
    if (a < b)
        min = a;
    else
        min = b;
    cout << "min = " << min << "\n";
    return 0;
}
```

Иҷроиши ин мисол чунин аст: аввал шarti $a < b$ санҷида мешавад, агар қимати a аз b хурд бошад, қимати он ба тағйирёбандаи `min` бахшида мешавад, вагарна (`else`) қимати b ба тағйирёбандаи `min` бахшида мешавад. Иҷроиши ин барнома ба оператории шarti (тернари) эквивалент мебошад, яъне `min=(a<b)?a:b`

Операторҳои шarti мураккаб(nested if-else)

Оператори `if` мутавонад ба намуди мураккаб иҷро гардад. Ин сохтори оператори шarti барои масъалаҳои, ки дорои шартҳои зиёд мебошанд истифода мешаванд. Тарзи истифодабарии ин оператор шакли зайлро дорад:

```
if (condition1) {
    Statements 1
}
Else if (condition2) {
    Statements 2
}
Else if (condition3) {
```

```

Statements 3
}
-----
Else if (conditionN) {
Statements N
}
Else (condition N+1){
Statements N+1
}

```

Мисол. Решаҳои муодилаи квадратии $ax^2 + bx + c = 0$ – ро ёбед.

Қайд. Маълум аст, ки $D = b^2 - 4ac$ буда,

1. ҳангоми $D < 0$ будан муодила ҳалли ҳақиқӣ надорад.
2. ҳангоми $D > 0$ будан решаҳои муодила аз рӯи формулаи зерин ҳисоб карда мешаванд.

$$x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$$

3. ҳангоми $D = 0$ будан решаҳои муодила бо ҳамдигар баробар буда, аз рӯи формулаи зерин ҳисоб карда мешаванд.

$$x_1 = -\frac{b}{2a} \text{ ва } x_2 = x_1$$

Матни барнома:

```

#include <iostream>
#include <cmath>
using namespace std;
int main(){
    float a, b, c, D, X1, X2;
    cout << "Adadhoi a,b va c - ro dohil kuned!\n";
    cin >> a >> b >> c;
    D = pow(b, 2) - 4 * a*c;
    if (D > 0)
    {
        X1 = (-b - sqrt(D)) / 2 * a;
        X2 = (-b + sqrt(D)) / 2 * a;
        cout << "D = " << D << " X1 = " << X1 << " X2 = " << X2 << "\n";
    }
    else if (D == 0)
    {
        X1 = -b / 2 * a;
        X2 = X1;
        cout << "D = " << D << " X1 = " << X1 << " X2 = " << X2 << "\n";
    }
    else if (D < 0)
    {
        cout << "Muodila reshahoi hakiki nadorad!\n";
    }
    return 0;
}

```

Тавре ки дар ин чо дидем, дар операторҳои шартӣ операторҳои мантиқӣ (logical operators) ё булӣ (Boolean operators) ва ё ифодаҳои, ки аз онҳо таркиб ёфтаанд ба сифати шарт (condition) истифода шудаанд. Операторҳои мантиқӣ хатман бо яке аз операторҳои шартии ==; >= ; <=; != ; > ; < истифода бурда мешаванд. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    int a;
    cin >> a;
    if (a > 10 && a < 15)
        cout << "Adeadi " << a << " dar porchai[10; 15] mehobad."
<< endl;
    return 0;
}
```

Мисол. Қимати ифодаи шартии зерин ҳисоб карда шавад.

$$f(x) = \begin{cases} \frac{x+y}{2xy}, & x \in [1; 4] \text{ ё } y < > 0 \\ 2xy, & \text{дар акси ҳол} \end{cases}$$

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
    double f, x, y;
    cout << "Adadhoi x va y - ro dohil kuned!\n";
    cin >> x >> y;
    if ((x >= 1 && x <= 4) || (y != 0 && x != 0))
        f = (x + y) / (2 * x*y);
    else
        f = 2 * x*y;
    cout << "f = " << f << "\n";
    return 0;
}
```

Оператори switch

Бигзор масъалае дорои якчанд шарт бошад ва талаб карда шавад, ки аз маҷмӯи шартҳои гузошташуда як ва ё якчанд шартӣ интихобшудааш иҷро карда шаванд. Барои иҷрои ин амал аз оператори switch истифода бурда мешавад. Агар дар оператори if-else ҳангоми дурустии шарт яке аз ду дастурҳои интихоб ва иҷро карда шаванд. Вале дар оператори switch бошад, аз маҷмӯи дастурҳои (шартҳои) танҳо дастури интихобшуда, ки ба шартӣ гузошташуда мувофиқ аст, иҷро карда мешавад. Наҳви умумии истифодаи ин оператор шакли зеринро дорад:

```
switch (expression)
{
    case 1: statement 1; break;
```

```

case 2: statement 2; break;
-----
default; statements; break;
}

```

Ин оператор чунин ичро карда мешавад. Агар қимати дохили қавси баъди switch омада (шарти гузошташуда) ба кадоме аз ин қиматҳо мувофиқ ояд, он гоҳ идоракунии ин барнома ба он сатр равона карда мешавад. Оператори мазкур фақат қимати бутун қабул мекунад. Дар ин сохтор case маънои интиҳоб, break маънои танаффус ва default бошад, маънои вагарнаро дорад.

Қайд: *Ҳамаи калимаҳои (case, break ва default) бояд пурра бо ҳарфи хурд навишта шаванд.*

Мисол. Барномае нависед, ки ҳангоми дохил намудани рақам номи онро барорад.

```

#include <iostream>
using namespace std;
int main(){
    int adad, i = 0;
    cout << "Adadro dohil kunedi!\n";
    while (i++ < 5)
    {
        cin >> adad;
        switch (adad) {
            case 1: cout << "one\n"; break;
            case 2: cout << "two\n"; break;
            case 3: cout << "thre\n"; break;
            case 4: cout << "four\n"; break;
            default: cout << "namedonam"; break;
        }
    }
    return 0;
}

```

Дар ин барнома тавре ки дида мешавад, оператории даврии while(i++<5) истифода шудааст. Оператори мазкур то ҳангоме ки қимати i ба 5 баробар шудан саҳифаи тасвириро дар ҳолати дохилкунӣ нигоҳ медорад. Дар ҳар як дохилкунӣ қимати i як воҳид афзоиш меёбад. Пас аз дохилкунӣ қимати adad санчида мешавад. Дар вақти санчиш қимати он ба кадом сатре, ки мувофиқ ояд, он сатр ичро мешавад. Яъне ҳангоми то ба чор баробар будани қимати adad дар саҳифаи тасвир мувофиқан one, two, thre, four ба чоп дода мешаванд. Дар ҳолате, ки қимати adad аз 4 боло ва ё аз як хурд бошад, калимаи «namedonam» ба чоп дода мешавад. Яъне, калимаи default маънои вагарна (ва ё case else)-ро дорад. Ба ғайр аз ин як оператор дар баъзе ҳолатҳо метавонад барои якчанд қимат ичро гардад.

Мисол. Барномае нависед, ки ҳангоми дохил кардани ҳарфҳо миқдори ҳарфҳои садонӣ ва ҳамсадорӣ ҳисоб кунад.

```

#include <iostream>
#include <cmath>
using namespace std;

```

```

int main(){
    char chr;
    int vowels = 0, consonants = 0, i = 0;
    cout << "Harfi anglisi dohil kuned\n";
    while (i++ < 6)
    {
        cin >> chr;
        switch (chr)
        {
            case 'a':
            case 'e':
            case 'i':
            case 'o':
            case 'u':
                vowels++;
                break;
            default:
                consonants++;
        }
    }
    cout << "Sadonocho = " << vowels << "\t" << "Hamsado= " << consonants
    << "\n";
    return 0;
}

```

Операторҳои switch ()-и дохилиҳам

Оператори switch() дар намуди дохилиҳам метавонад низ иҷро гардад. Агар доимие, ки барои иҷроиши оператори switch() дар оператори switch()-и берунӣ гузошта шудааст, ба доимии switch()-и дохилӣ якхела бошад низ барнома ба хатогӣ рӯбуруӣ намешавад. **Мисол.:**

```

#include <iostream>
using namespace std;
int main(){
    int m, n;
    cout << "n = "; cin >> n;
    cout << "m = "; cin >> m;
    switch (n) {
        case 1: cout << "Adadi 1 az switchi beruni ast!" << endl;
            switch (m) {
                case 1: cout << "Adadi 1 az switchi dohili ast!" << endl;
                    break;
                case 2: cout << "Adadi 2 az switchi dohili ast!" << endl;
                    break;
            }
            break;
        case 2: cout << "Adadi 2 az switchi beruni ast!" << endl;
            break;
    }
    return 0;
}

```


Тавре ки дида мешавад, дар ин ҷо ду оператори swith() дар намуди дохилиҳам омадаанд. Қимати шартҳои ҳар ду якхела мебошанд. Забони С++ ба зудӣ метавонад байни онҳо фарқ гузорад. Натиҷаи барнома шакли зеринро дорад:

Операторҳои такрорӣ

Дар табиат ҳама гуна ҳодисаҳо ба намуди даврӣ иҷро мешаванд. Масалан, гардиши шабонарӯз, акрабаки соҳат, Замин дар атрофии Офтоб ва ғайра. Дар забонҳои барномасозӣ низ барои ҳалли масъалаҳои хусусияти даврӣ дошта, ба монанди ҳисоб намудани суммаҳо, факториали адад, якчанд маротиба иҷро намудани ягон қисмати барнома ё дастур ва ғайра операторҳои даврӣ вучуд доранд. Ин операторҳо имконияти якчанд маротиба иҷроиши як дастур ва ё як қитъаи барномаро фароҳам меоранд. Пайдарпайии дастурҳое, ки ҳангоми дар дохили давр иҷро мешаванд, тани давр номида мешавад. Як маротиба иҷроиши операторро як қадам ё итератсия меноманд. Шерте, ки дар аввал гузошта шудааст ва ба охири расиши даврро муайян мекунад, параметри давр номида мешавад. Муҳимтарин ва ё асоситарин операторҳои даврӣ дар забони С++ се то мебошанд, ки дутоашонро операторҳои даврӣ бо шартӣ пешоянду пасоянд ва яктоашро оператори даврии параметрдор меноманд.

Оператори даврӣ бо шартӣ пешоянд

Дар оператори даврӣ бо шартӣ пешоянд шартӣ баохиррасиши давр дар аввал гузошта мешавад ва чанд маротиба такрор шудани ин оператор дар аввал муайян нест ва то шартӣ гузошташуда ҳақ будан иҷро карда мешавад. Аввал шартӣ гузошташуда санҷида мешавад, баъд операторҳои дохили ҳалқа иҷро мешаванд. Тарзи истифодабарии ин оператор шакли зайлро дорад:

Тарзи истифодаи оператор	Тарзи истифодаи оператор
while(condition) { Statements }	while(шарт) { мачмӯи операторҳо }

Мачмӯи операторҳои дохили ҳалқа то он замоне такроран иҷро мешаванд, ки шартӣ гузошташуда ҳақ бошад. Намуди умумии иҷроиши оператори мазкур ба намуди зерин иҷро мешавад: Қимати операторҳои дохилии ҳалқа як маротиба иҷро гардида, шартӣ гузошташуда санҷида мешавад агар он ҳақ бошад, мачмӯи операторҳо аз нав иҷро карда мешаванд, то он замоне ки шартӣ гузошташуда ҳақ бошад. Шартӣ гузошташуда пеш аз ҳар як итератсия санҷида мешавад. Чанд итератсия иҷро шудани давр дар ибтидо маълум нест.

Мисол. Ҳамаи ададҳои натуралии аз як то 20-ро дар саҳифаи тасвир бароред.

```
#include <iostream>
using namespace std;
int main(){
    int i = 0;
```

```

while (i < 20)
{
    ++i;
    cout << i << "\t";
}
return 0;
}

```

Дар ин ҷо `i++` ва `cout<<i<<«\t»` операторҳои дохили давр буда, ифодаи `i++` пас аз ҳар итератсия ба `i` як воҳид илова намуда, шарти гузошташударо месанҷад, агар қимати `i` ба 20 баробар шавад, дигар иҷроиши давр боздошта шуда, идоракунӣ ба операторҳои баъдидаврӣ мегузарад.

Мисол. Ду адади бутун дода шудаанд КТУ-и ин ададхоро ёбед.

```

#include <iostream>
using namespace std;
int main(){
    int adad1, adad2, S, KTU;
    cout << "Adadi yakum va duyum - ro vorid kuned!\n";
    cin >> adad1 >> adad2;
    while (S > 0)
    {
        S = adad1%adad2;
        adad1 = adad2;
        adad2 = S;
        KTU = adad1;
    }
    cout << "KTU = " << KTU << "\n";
    return 0;
}

```

Оператори даврии бо шарти пасоянд `do/while`

Оператори `do/while` ҳам монанди оператори `while` амал мекунад, вале дар ин оператор аввал операторҳои дохили ҳалқа иҷро гардида, баъд шарти гузошташудак санҷида мешавад. Аз ин рӯ, оператори мазкурро оператори даврии бо шарти пасоянд меноманд. Тарзи истифодабарии ин оператор чунин аст:

```

do
{
    Statements-Маҷмӯи операторҳои дохили давр
}
while (condition-шарт)

```

Мисол. Мисоли пешинаро бо ёрии оператории даврии бо шарти пасоянд иҷро кунед.

```

#include <iostream>
using namespace std;
int main(){
    int i = 0;
    do
    {
        i++;

```

```

        cout << i << "\t";
    } while (i < 20);
    return 0;
}

```

Мисол. Суммаи ҳамаи ададҳои ҷуфти порчаи $[0...20]$ -ро ҳисоб кунед.

```

#include <iostream>
using namespace std;
int main(){
    int i = 0, Summa = 0;
    do
    {
        i += 2;
        Summa += i;
    } while (i < 20);
    cout << "Summai adadhoi jufti az 0 to 20 barobar ast ba : " << Summa
    << "\n";
    return 0;
}

```

Қайд. Дар ин мисол як тағйирёбанда ба номи *Summa* муаррифӣ карда шудааст ва ба он ба сифати қимати ибтидоӣ 0 бахшидем. Ифодаи $i+=2$ -қимати i -ро дар ҳар як итератсия ду воҳид зиёд мекунад, то замоне ки қимати i ба 20 баробар шавад. Ифодаи $Summa+=i$ бошад, дар ҳар зиёдшавии i қимати i -ро ба *Summa* мебахшад ва дар охир қимати *Summa* ба ҷоп дода мешавад.

Оператори даврии параметрдор

Оператори даврии параметрдор операторе мебошад, ки дар он оператор қимати ибтидоӣ, интиҳой ва қадами зиёдшавии давр нишон дода мешавад. Тарзи истифодабарии ин оператор шакли зайлро дорад:

```

for(variable; shart; change-var)
{
    statements
}

```

Дар ин ҷо *variable* ифодае мебошад, ки ба тағйирёбанда (пара-метр)-и давр қимати аввала мебахшад, *change-var* –ифодаест, ки қимати *variable*-ро тағйир медиҳад ва *shart* бошад, ифодаеаст, ки шарт баитмомрасии даврро муайян мекунад. Ба таври дигар гӯем, *variable*-як ифодаи бахшиш буда, ба тағйирёбандаи давр қимати аввала мебахшад, *shart* –ифодаи мантиқиест, ки шарт баитмомрасии даврро муайян мекунад ва *change-var* бошад, як ифодаи унари (ягона) ва ё бахшиш буда, қимати тағйирёбандаи даврро тағйир медиҳад.

Оператори мазкур монанди оператори *while* амал мекунад, яъне пас аз ҳар итератсия шарт гузошташуда санҷида мешавад ва то замоне ки он ҳақ аст, давр такрор меёбад.

Мисоли1. Суммаи ҳамаи ададҳои натуралии аз 0 то 20-ро ҳисоб кунед.

```

#include <iostream>
using namespace std;
int main(){
    int i, Summa = 0;

```

```

    for (i = 0; i < 20; i++)
        Summa += i;
cout << "Summai adadhoi naturalii az 0 to 20 barobar ast : " << Summa
<< "\n";
    return 0;
}

```

Қайд. Дар ин барнома ду тағйирёбанда аз навъи бутун *Summa* ва *i* муаррифӣ карда шудааст. Синас, оператори *for (i=0;i<20;i++)*, ки дорои се параметр мебошад, иҷро карда мешавад. Параметри якум ба тағйирёбандаи *i* қимати ибтидоии нолӣ мебахшад. Параметри дуввум ифодаи шартии *i<20* ҳалқаро маҷбур месозад, ки то қимати *i* баробари 20 шудан, иҷро гардад. Параметри сеюм бошад, ифодаи унари *i++* қимати *i*-ро дар ҳар итератсия як воҳид афзоиш медиҳад. Ин барномаро метавонем низ ба суратҳои зерин нависем.

1)

```

#include <iostream>
using namespace std;
int main(){
    int i = 0, Summa = 0;
    for (; i < 20; i++)
        Summa += i;
cout << "Summai adadhoi naturalii az 0 to 20 barobar ast : " << Summa
<< "\n";
    return 0;
}

```

2)

```

#include <iostream>
using namespace std;
int main(){
    int i = 0, Summa = 0;
    for (; i < 20;)
        Summa += i++;
cout << "Summai adadhoi naturalii az 0 to 20 barobar ast : " << Summa
<< "\n";
    return 0;
}

```

3)

```

#include <iostream>
using namespace std;
int main(){
    int Summa, i;
    for (Summa = 0, i = 0; i < 20; i++)
        Summa += i;
cout << "Summai adadhoi naturalii az 0 to 20 barobar ast : " << Summa
<< "\n";
    return 0;
}

```

4)

```

#include <iostream>
using namespace std;
int main(){
    int Summa = 0, i = 0;
    for (;;) {
        Summa += i;
        i++;
        if (i >= 20) break;
    }
    cout << "Summai adadhoi naturalii az 0 to 20 barobar ast : " << Summa
    << "\n";
    return 0;
}

```

Тавре ки дидем, як оператори for-ро ба чандин тарз истифода бурдан мумкин аст. Аз ин ҷо чунин хулоса мебарояд, ки барои компилятор фаҳмо будани параметр аст на тарзи навишти он. Дар мисоли сеюм бошад, оператори коммо истифода шудааст ва он нишон медиҳад, ки дар дохили қавси ин оператор якчанд ифода навишта мешавад ва онҳо аз ҳамдигар бо оператори мазкур чудо карда шудаанд.

Сохтори даврҳои мураккаб ё дохилиҳам

Дар дохили як ҳалқа ба сифати операторҳои дохилӣ, яъне statements метавонем ҳалқаи дигарро нависем, вале дар ин навишт бояд диққат дод, ки ҳалқаҳо ҳамдигарро маҳдуд накунад. Дар ин на-муд операторҳо аввал оператори берунӣ иҷро мешавад, баъд операторҳои дохилӣ.

Мисол. Даврҳои дохилиҳам.

```

#include <iostream>
using namespace std;
int main(){
    int i, j;
    for (i = 1; i <= 3; i++)
        for (j = 1; j <= 4; j++)
            cout << i << "\t";
    return 0;
}

```

Барномаи мазкур ҳар як қимати i- дар ҷои худ 4 маротибагӣ чоп мекунад. Натиҷаи барнома намуди зеринро дорад:

1	1	1	1	2	2	2	2	3	3	3	3
---	---	---	---	---	---	---	---	---	---	---	---

Барномаи зерин бошад, ҳар маротиба қиматҳои j-ро 3 маротиба ба чоп медиҳад.

```

#include <iostream>
using namespace std;
int main(){
    int i, j;
    for (i = 1; i <= 3; i++)
        for (j = 1; j <= 4; j++)
            cout << j << "\t";
}

```

```

return 0;
}

```

Натиҷаи ин барнома ба сурати зерин аст:

1	2	3	4	1	2	3	4	1	2	3	4
---	---	---	---	---	---	---	---	---	---	---	---

Мисол. Барномае нависед, ки чадвали зарбро дар саҳифаи тасвир ба чоп диҳад.

```

#include <iostream>
using namespace std;
int main(){
    int i, j;
    for (i = 1; i <= 10; i++)
    {
        for (j = 1; j <= 10; j++)
            cout << i*j << "\t";
        cout << endl;
    }
    cout << endl;
    return 0;
}

```

Ин барнома як қимати *i*-ро интихоб карда, ба ҳамаи қиматҳои *j* зарб мекунад, то он замоне ки шартҳои гузошташуда дуруст (true- ҳақ) бошанд. Натиҷаи ин барнома ба сурати зерин аст:

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

Мисол. Барномае нависед, ки ададҳои натуралии аз 1 то 15-ро ба сурати зерин чоп кунад.

```

1 4 7 10 13 16 19 22 25 28
2 5 8 11 14 17 20 23 26 29
3 6 9 12 15 18 21 24 27 30
4 7 10 13 16 19 22 25 28 31
5 8 11 14 17 20 23 26 29 32

```

Матни барнома:

```

#include <iostream>
using namespace std;
int main(){
    int i, j;
    for (i = 1; i <= 5; i++){
        for (j = 1; j <= 10; j++)
            cout << 3 * (j - 1) + i << " ";
    }
}

```

```

        cout << endl;
    }
    cout << endl;
    return 0;
}

```

Мисол. Барномае нависед, ки ададҳои ба панҷ каратиرو аз поён ба боло ва ададҳои каратии дахро аз боло ба поён чоп кунад.

```

#include <iostream>
using namespace std;
int main(){
    int i, j;
    for (i = 5, j = 250; i < j; i += 5, j -= 10)
        cout << i << " " << j << endl;
    cout << endl;
    return 0;
}

```

Оператори коммо (the comma operators)

Тавре ки қаблан гуфтем, оператори коммо операторе мебошад, ки ба ҷои як ифода имконияти иҷрошавии якчанд ифодаро фароҳам меорад. Ин оператор бештар бо оператори for истифода мегардад ва он имконият медиҳад, ки якчанд ифодаи дар дохили қавси баъди оператори for омада иҷро гарданд.

Мисол: Суммаи n адади аввалаи натуралиро ҳисоб кунед.

```

#include <iostream>
using namespace std;
int main(){
    int Summa, i, n;
    for (cout << "n = ", cin >> n, Summa = 0, i = 0; i <= n; Summa =
Summa + i, i = i + 1);
    cout << "Summa = " << Summa << "\n";
    return 0;
}

```

Тавре ки мушоҳида кардем, оператори коммо иҷозат дод, ки ба ҷои як expression якчанд expression-иҷро гардид. Тарзи истифодабарии ин оператор чунин аст:

```
for (expression1a, expression1b; expression1; expression2) statements
```

Дар ин ҷо expression1a, expression1b ифодаҳои ҳастанд, ки аз ҳамдигар бо ёрии оператори коммо ҷудо карда шудаанд.

Ин оператор аз ҷап ба рост амал мекунад, яъне иҷрои амалҳо аз ҷап ба рост аст. Ҳангоми муаррифӣ намудани тағйирёбандаҳо оператори коммо танҳо онҳоро аз ҳамдигар ҷудо мекунад. Дар забони C++ оператори мазкур ҳамчун оператори сатҳи поён шинохта мешавад.

Операторҳои continue ва break

Операторҳои continue ва break операторҳои мебошанд, ки ҷараёни барномаро қатъ мекунанд. Оператори break операторе мебошад, ки идоракунии

(контролӣ) барномаро ба поёнии сохтаре, ки break дар дохили он аст, интиқол медиҳад, яъне ин оператор сабаби баромадан аз барнома мегардад.

Тавре ки аз мисолҳои боло дидем, ин оператор идоракунии ба сатри навбатӣ ва ё ба сатре, ки қимате он ба қимати додашуда мувофиқ меояд, интиқол медиҳад. Аслан ин оператор бо операторҳои даврии while, do/while ва for истифода мешавад ва сабаби баромадан аз давр мешавад. Мисол:

```
#include <iostream>
using namespace std;
int main(){
    int i = 0;
    while (++i < 20)
    {
        if (i == 10)
            break;
        cout << i << "\t";
    }
    cout << "\nHalka hangomi i = 9 budan qat meshavad.\n";
    return 0;
}
```

Аз матни барнома айён аст, ки ҳангоми $i < 9$ будан қимати i -ро ба чоп медиҳад ва оператори break қиматҳои дигарро сарфи назар мекунад. Оператори мазкур сабаби баромадан аз дохили оператори даврӣ ва ё шартӣ мегардад.

Оператори continue (идома) бошад операторест, ки барои гузаштан аз як қисмати барнома ба қисмати дигари он истифода бурда мешавад, яъне ин оператор аз такрори ҷорӣ давр як ҳалқаро сарфи назар мекунад ва ё як ҳалқаро нодида мегузарад. Ин оператор ҳам дар операторҳои даврии while, do/while ва for истифода мешаванд.

Мисол. Истифодаи оператори continue бо оператори даврии while

```
#include <iostream>
using namespace std;
int main(){
    int i = 0;
    while (++i < 20){
        if (i == 10)
            continue;
        cout << i << "\t";
    }
    cout << "\nHalka qimati i = 9-ro sarfi nazar mekunad.\n";
    return 0;
}
```

Чунончи мушоҳида гардид, оператори continue дар оператории даврии while як итератсияро, яъне $i=9$ -ро сарфи назар мекунад, яъне онро нодида мегузарад.

Оператори гузариши бешарт (goto)

Оператори goto барои гузаштан аз як қисмати барнома ба қисмати дигар ба кор бурда мешавад. Ҳангоми истифодаи оператори мазкур барои гузаштан

ба мавқеи лозимӣ бояд нишонае истифода бурда шавад. Нишона метавонад ягон калимаи ёрирасон ва ё калимаи дигар бошад. Нишона ҳатман бо ҳарф ва ё рамзи хати поён сар мешавад. Тарзи истифодабарии ин оператор чунин аст:

goto label, ки дар ин ҷо label нишонаест, ки дар аввали ягон сатри барнома навишта шудааст. Бояд пас аз он ҳатман ду нуқта гузошта шавад. Мисол.

```
{
  :
  goto end;
  :
}
end: cout<<"Barnamea ba ohir
rasid!\n";
```

Бояд қайд кард, ки монанди забони Паскал дар забони C++ нишона (Label)-е ки бояд идоракуни ба он гузашта шавад, муаррифӣ карда намешавад.

Мисол. Барномае нависед, ки суммаи n адади аввалаи натуралиро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int main(){
    int Summa = 0, i = 0, n;
    cout << "n = ";
    cin >> n;
label:
    Summa += i;
    i++;
    if (i <= n) goto label;
    cout << "Summa = " << Summa << "\n";
    return 0;
}
```

Қайд. Пас аз иҷрои ҳар як давр қимати i санҷида мешавад, агар шарт дуруст бошад, идоракуни ба сатре, ки оператори goto ба он ишора кардааст (дар ин мисол label), мегузарад. Ва бо ҳамин монанд ин амал то замоне ки шarti гузошташуда дуруст аст, такрор мегардад.

Саволҳо барои мустаҳкамкунӣ

1. Операторҳои идоракуни чӣ гуна оператор мебошанд?
2. Оператори while аз do/while чӣ фарқ мекунад?
3. Оператори даврии параметрдор чӣ гуна оператор аст?
4. Операторҳои мураккаб чӣ гуна оператор мебошанд?
5. Оператори goto чӣ гуна оператор аст?
6. Оператори continue бештар дар кадом оператор истифода бурда мешавад?
7. Оператори break чӣ амалро иҷро мекунад?
8. Оператори continue аз оператори break бо чӣ фарқ мекунад?
9. Операторҳои мантиқӣ дар кадом оператор бештар истифода

- мешаванд?
10. Оператори case чӣ гуна оператор аст?
 11. Магар мумкин аст, ки дар як мисол ҳам оператори while ва ҳам оператори do ... while истифода бурда шавад?
 12. Операторҳои continue ва break метавонанд бо операторҳои шартӣ якҷоя истифода шаванд?
 13. Оё истифодаи continue ва break зарур аст?
 14. Ҳалқаҳои дохилиҳам барои чӣ лозиманд ва дар як барнома чанд ҳалқа метавонанд дохилиҳам бошанд?
 15. Дар дохили оператори case магар мумкин аст, ки оператори шартии if истифода бурда шавад?

Намунаи ҳалли масъалаҳо

1. Барномае нависед, ки амали калкуляторро иҷро намояд.

```

#include <iostream>
#include <cmath>
using namespace std;
int main(){
    double x, p;
    char op;
    cout << "\n >> ";
    cin >> x;
    p = x;
    while (1) {
        cout << "\n >> ";
        cin >> op;
        if (op == '=' ) {
            cout << "---->" << p << endl;
            exit(0);
        }
        cout << " >> ";
        cin >> x;
        switch (op) {
            case '+': p += x; break;
            case '-': p -= x; break;
            case '*': p *= x; break;
            case '/': p /= x;
        }
        cout << p;
    }
    return 0;
}

```

2. Адади натуралии N сода номида мешавад, агар он фақат ба худаш ва ба як тақсим шавад. Барномае нависед, ки миқдори ададҳои содаи фосилаи [2; N]-ро ҳисоб кунад ва ададҳои содаро ба чоп диҳад.

```

#include <iostream>
using namespace std;
int main(){

```

```

int i, j, N;
bool d;
cout << "N = "; cin >> N;
for (i = 2; i <= N; i++)
{
    d = false;
    for (j = 2; j <= i / 2; j++)
        if (i%j == 0)d = true;
    if (!d) cout << i << "\t";
}
return 0;
}

```

3. Адади N мукамал номида мешавад, агар он ба суммаи тақсимкунандаҳош ғайр аз худаш баробар бошад. Масалан, адади 6 мукамал аст, чунки $6=1+2+3$. Барномае нависед, ки ҳангоми дохил намудани адад агар мукамал бошад, сатри «мукамал». Ва худи ададро ба чоп диҳад.

```

#include <iostream>
using namespace std;
int main(){
    int i, N, S = 1;
    cout << "N = "; cin >> N;
    for (i = 2; i <= N / 2; i++)
        if (N%i == 0) S += i;
    if (S == N)
        cout << S << " Addadi mukammal meboshad : " << endl;
    return 0;
}

```

4. Якчанд адади натуралӣ пасиҳам ворид карда мешаванд. Адади 0 охири ин пайдарпайи мебошад. Аз ин ададҳои додашуда калонтаринашро бароред. Инчунин миқдори ададҳои калонтаринро бароред.

```

#include <iostream>
using namespace std;
int main(){
    int N, Max, k;
    for (cout << "N = ", cin >> N, k = 0; N != 0; cout << "N = ", cin >> N)
        if (k == 0) {
            k = 1; Max = N;
        }
        else if (N > Max) {
            Max = N;
            k = 1;
        }
        else if (N == Max) k++;
    if (k)
        cout << "Max = " << Max << "\n" << "k = " << k << "\n";
    return 0;
}

```

5. Якчанд адади натуралӣ пасиҳам ворид карда мешаванд. Адади 0 охири ин пайдарпайи мебошад. Барномае нависед, ки қимати миёнаи арифметикии ин ададхоро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int main(){
    float N, k, S = 0, MArif;
    for (cout << "N = ", cin >> N, k = 0; N != 0; cout << "N = ", cin
    >> N, k++)
        S += N;
    MArif = S / k;
    cout << "k = " << k << "\t" << "MArif = " << MArif;
    return 0;
}
```

6. Барномае нависед, ки бо дохил кардани таърихи рӯз, рӯзи баъдинаро барорад. Масалан :

Дохилкунӣ: 31 12 1989

Хориҷкунӣ: 1 1 1990

```
#include <iostream>
using namespace std;
int main(){
    int ruz, moh, sol, k = 0;
    cout << "Ruz moh sol\n";
    cin >> ruz;
    cin >> moh;
    cin >> sol;
    if (moh == 2) {
        if ((sol % 4) != 0 && ruz == 28) k = 1;
        if ((sol % 4) == 0 && ruz == 29) k = 1;
    }
    else if ((moh == 4 || moh == 6 || moh == 9 || moh == 11) && (ruz
    == 31))
        k = 1;
    else if (ruz == 31) k = 1;
    if (k == 1) {
        cout << "ogohi: Ohiri ruzi moh ast!\n";
        ruz = 1;
        if (moh == 12) {
            moh = 1;
            sol++;
            cout << "Soli nav omad, shodu gazalhon.\n";
        }
        else moh++;
    }
    else ruz++;
    cout << ruz << " " << moh << " " << sol << "\n";
    return 0;
}
```

7. Адади N ба намуди дахӣ дода шудааст. Барномае нависед, ки онро ба намуди дуй гардонад.

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
    int N, k, b = 1; unsigned long summa = 0;
    cout << "N = "; cin >> N;
    while (N) {
        k = N % 2;
        k = k*b;
        summa += k;
        b = b * 10;
        N /= 2;
    }
    cout << "N / 2 = " << summa << "\n";
    return 0;
}
```

8. Барномае нависед, ки қимати суммаи дукаратаро ҳисоб кунад.

$$\sum_{i=1}^m \sum_{j=1}^n (i + j)$$

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
    int i, j, n, m;
    float S = 0;
    cout << "n = "; cin >> n;
    cout << "m = "; cin >> m;
    for (i = 1; i <= m; i++)
        for (j = 1; j <= n; j++)
            S += i + j;
    cout << "S = " << S << "\n";
    return 0;
}
```

9. Барномае нависед, ки бо дохил кардани таърихи рӯз номи рӯзро барорад. Масалан:

Дохилкунӣ: 25 04 1989

Хориҷкунӣ: Сешанбе

Қайд. Барои ёфтани рақами рӯзҳои ҳафта бо дода шудани таърихи рӯз аз формулаи

$$\left(d + \left[\frac{1}{5} (13m - 1) \right] + y + \left[\frac{y}{4} \right] + \left[\frac{c}{4} \right] - 2c + 777 \right) \bmod 7,$$

истифода бурда мешавад, ки дар ин ҷо d -рӯзи ҳафта, m – рақами моҳ, c , y рақами аср ва сол дар ин аср буда, $[\]$ -ишораи қисми бутун мебошад. Инчунин рӯзҳои ҳафта чунин рақамгузорӣ мебошанд.

0-Якшанбе, 1-душанбе,..., 6-шанбе

```
#include <iostream>
using namespace std;
int main(){
    int ruz, moh, sol;
    int c, y, m, d;
    cin >> ruz >> moh >> sol;
    if (moh == 1 || moh == 2)
        sol--;
    m = moh - 2;
    if (m <= 0) m += 12;
    c = sol / 100;
    y = sol - c * 100;
    d = (ruz + (13 * m - 1) / 5 + y + y / 4 + c / 4 - 2 * c + 777) % 7;
    switch (d) {
        case 1:puts("Dushanbe"); break;
        case 2:puts("Seshanbe"); break;
        case 3:puts("Chorshanbe"); break;
        case 4:puts("Panjshanbe"); break;
        case 5:puts("Juma"); break;
        case 6:puts("Shanbe"); break;
        case 0:puts("Yakshanbe"); break;
    }
    return 0;
}
```

12. Адади 5 рақама дода шудааст, муайян кунед, ки оё он ба 5 тақсим мешавад. Агар тақсим шавад, натиҷаи тақсимо бароред. Ва дар акси ҳол пайғоми «намешавад» хориҷ карда шавад.

```
#include <iostream>
using namespace std;
int main(){
    int N;
    bool bl;
    cout << "N = "; cin >> N;
    bl = (N % 5);
    if (bl)
        cout << "Adadi dodashuda ba 5 taqsim nameshavad\n";
    else
        cout << "N / 5 = " << N / 5;
    return 0;
}
```

10. Ҳисоб кунед: $S = \sqrt{3 + \sqrt{5 + \sqrt{7 + \dots + \sqrt{99}}}}$

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
```

```

float S = 0;
int i = 99;
while (i >= 3){
    S = sqrt(S + i);
    i -= 2;
}
cout << "S = " << S << " \n ";
return 0;
}

```

Масъалаҳо барои кори мустақилона Масъалаҳои гурӯҳи If

1. Ду адади ҳақиқӣ дода шудааст. Агар онҳо ба ҳамдигар баробар набоянд, он гоҳ бо ҳам зарб карда шаванд. Дар ҳолати акс ба онҳо қимати сифр бахшида шавад.
2. Се адади ҳақиқӣ дода шудааст. Хурдтарини онҳо ёфта шавад.
3. Се адади ҳақиқӣ дода шудааст. Аввал хурдтарини онҳо ва баъд калонтарини онҳо ёфта шавад.
4. Се адади ҳақиқӣ дода шудааст. Сумма ду адади хурдтарин ёфта шавад.
5. Се адади ҳақиқӣ дода шудааст. Адади миёна (ададе ки дар байни ададҳои хурдтарин ва калонтарин ҷойгир аст) хориҷ карда шавад.
6. Се нуқта бо координатаҳояшон дода шудаанд, муайян кунед, ки оё онҳо қуллаҳои секунҷаро ифода мекунанд ё не.
7. Ду суръат яке бо километр/соат ва дигаре бо метр/сония дода шудаанд. Қадоме аз ин суръатҳо тезтар мебошад?
8. Ҳаҷм ва вазни ду қисми аз моддаҳои гуногун сохташуда дода шудааст. Қадоме аз ин ду қисмҳо зичии зиёд дорад?
9. Ададҳои ҳақиқии a, b, c ($a \neq 0$) дода шудаанд. Решаҳои муодилаи $ax^2 + bx + c = 0$ ёфта шаванд.
10. Масоҳати квадрат ва доира дода шуда аст. Муайян карда шавад:
 - а) доира дар дохили квадрат ҷой мегирад ё не?
 - б) квадрат дар дохили доира ҷой мегирад ё не?
11. Ду адади бутуни m ва n дода шудаанд. Агар адади m ба адади n бебақия тақсим шавад натиҷаи тақсим ва дар ҳолати акс иттилооти матнии «адади m ба адади n бебақия тақсим намешавад» бароварда шавад.
12. Адади дурақамаи n дода шудааст. Муайян карда шавад:
 - а) оё суммаи рақамҳои он адади дурақама мешавад ё не?
 - б) оё он аз суммаи рақамҳояш калон аст ё не?
13. Адади натуралии серақама дода шудааст. Муайян карда шавад:
 - а) оё рақами дуюми он сифр аст?
 - б) оё рақами якуми он тоқ аст?
14. Муайян карда шавад, ки секунҷаи бо тарафҳои додашудаи a, b, c секунҷаи баробартараф мебошад ё не.

15. Ададҳои мусбати ҳақиқии a, b, c, d дода шудаанд. Муайян карда шавад, ки оё роткунҷаи бо тарафҳои a ва b бударо дар дохили росткунҷаи бо тарафҳои c ва d буда ҷойгир кардан мумкин аст ё не.
16. Координатаҳои нуқта дар ҳамворӣ дода шудаанд. Дар кадом чоряк ҷойгир будани онро муайян кунед.
17. Чор адади бутун дода шудааст. Миқдори адади чуфти дар байни онҳо мавҷуд буда муайян карда шавад.
18. Бо дода шудани сол. Миқдори рӯзҳо дар он сол ҳисоб карда шавад. Сол аз 366 шабонарӯз иборат аст, агар он бе бақия ба 4 тақсим шавад, ба истиснои солҳое, ки ба 100 тақсим мешаванду ба 400 не. Дар акси ҳол аз 365 шабонарӯз иборат мебошад.
19. Се адад дода шудааст. Миқдори ададҳои манфӣ ва мусбат аз ин набор муайян карда шавад.
20. Соати акрабақдор дар ҳар соат ду дақиқа ақиб мемонад. Агар ҳамин ақибмонӣ идома ёбад, баъд аз кадом муддати камтарини вақт соат боз аз нав вақти аниқро нишон хоҳад дод?

Масъалаҳои гурӯҳи Case

1. Ҳангоми дохил кардани рақами тартибии моҳ (1,2,...,12) номи моҳ (январ, феврал,..., декабр) ва фасли сол (зимистон, баҳор, тобистон, тирамоҳ) хориҷ карда шавад.
2. Амалҳои арифметикӣ бо ададҳо чунин рақамгузорӣ шудаанд: 1-чамъ, 2-тарҳ, 3-зарб, 4- тақсим. Рақами иҷроиши амал ва ду адади бутуни мусбати ғайрисиғфрӣ дода мешавад. Амали интиҳобшуда бо ин ададҳо иҷро гашта натиҷа ба чоп бароварда шавад.
3. Рӯзҳои ҳафта чунин рақамгузори шудаанд: 1-душанбе, 2-сешанбе,..., 6-шанбе, 7-якшанбе. Адади бутуни K -и фосилаи 1-365 ҷойгирбуда дода шудааст. Номи рӯзи ҳафтaro барои рӯзи K —уми сол муайян кунед, агар маълум бошад, ки якуми январӣ ин сол сешанбе аст.
4. Қисми ҳазорӣ, садӣ, даҳӣ ва воҳиди адади чоррақама бо тартиби муайян бо ададҳои 1, 2, 3, 4 рақамгузорӣ гаштааст. Дар ҳолати дохил намудани адади чоррақамаи ихтиёрӣ ва дохил намудани яке аз ададҳои муайянкардашуда, қисми интиҳобшудаи адади чоррақама ба чоп бароварда шавад.
5. Дар як китобхона рӯзи панҷшанбеи охири ҳар моҳ рӯзи санитарӣ мебошад. Адади натуралии n , ки рақами солро ифода менамояд, дода шудааст. Таърихи ҳамаи рӯзхоеро, ки дар моҳҳои январ, феврал, ..., декабри ин сол рӯзи санитарӣ ба онҳо мувофиқ меояд, ба тартиб бароред.
6. Рӯзҳои ҳафта чунин рақамгузори шудаанд: 1-душанбе, 2-сешанбе, ..., 6-шанбе, 7-якшанбе. Адади бутуни K -и фосилаи 1-365 ҷойгирбуда дода шудааст. Номи рӯзи ҳафтaro барои рӯзи K -уми сол муайян кунед, агар маълум бошад, ки якуми январӣ ин сол шанбе аст.
7. Рӯзҳои ҳафта чунин рақамгузори шудаанд: 1-душанбе, 2-сешанбе, ..., 6-шанбе, 7-якшанбе. Адади бутуни K -и фосилаи 1-365 ҷойгирбуда дода шудааст. Номи рӯзи ҳафтaro барои рӯзи K -уми сол муайян кунед, агар маълум бошад, ки якуми январӣ ин сол чоршанбе аст.

8. Рӯзҳои ҳафта чунин рақамгузори шудаанд: 1-душанбе, 2-сешанбе, ..., 6-шанбе, 7-якшанбе. Ададҳои бутуни $K \in [1,365]$ ва $m \in [1,7]$ дода шудааст. Номи рӯзи ҳафтaro барои рӯзи K -уми сол муайян кунед, агар маълум бошад, ки якуми январи ин сол рӯзи m аст.
9. Координатаҳои майдони тахтаи шохмот x, y (ададҳои бутуни дар фосилаи 1-8 хобанда) дода шудааст. Сиёҳ будани майдони поёнии чапи тахта (1x1)-ро ба назар гирифта, муайян кунед, ки оё майдони (x, y) сафед аст?
10. Координатаҳои ду майдони тахтаи шохмот (x, y) ва (x_1, y_1) (ададҳои бутуни дар фосилаи 1-8 хобанда) дода шудааст. Сиёҳ будани катакчаи чапи поёнии тахта (1x1)-ро ба назар гирифта, муайян кунед, ки оё ранги майдонҳои (x, y) ва (x_1, y_1) якхела аст?
11. Ду адади бутуни D (рӯз) ва M (моҳ), ки санаи соли ғайрикабисаро муайян мекунад, дода шудааст. Қиматҳои D ва M -ро барои санаи ба санаи додашуда гузаштаро муайян кунед.
12. Ду адади бутуни D (рӯз) ва M (моҳ), ки санаи соли ғайрикабисаро муайян мекунад, дода шудааст. Қиматҳои D ва M -ро барои санаи ба санаи додашуда баъдинаро муайян кунед.
13. Элементҳои секунҷаи росткунҷаи баробарпахлӯ чунин рақамгузори гаштаанд: 1- катет a , 2 – гипотенуза $c = a\sqrt{2}$, 3 – баландӣ h , ки ба гипотенуза фароварда шудааст ($h=c/2$), 4 – масоҳат $S=c \cdot h/2$. Рақами яке аз ин элементҳо бо қиматаш дода шудааст. Қимати дигар элементҳои секунҷа ёфта шавад.
14. Элементҳои давра чунин рақамгузори шудаанд: 1-радиус R , 2-диаметр, $D=2R$, 3-дарозӣ, $2L = 2\pi R$, 4-масоҳати давра, $S = \pi R^2$. Рақами яке аз ин элементҳо ва қимати он дода шудааст, қимати элементҳои боқимондаро ҳисоб кунед.
15. Адади бутуни k ($1 \leq k \leq 180$) ва пайдарпаии ададҳои дурақама бо намуди 10111213...9899 дода шудаанд. Номи адади k -ум муайян карда шавад.
16. Адади бутуни k ($1 \leq k \leq 222$) ва пайдарпаии ададҳои дурақама ба намуди 123...91011..9899100101...109110 дода шудаанд. Номи адади k -ум муайян карда шавад.
17. Кори чароғак барои ронандагон чунин барномасозӣ карда шудааст: аз оғози лаҳзаи ҳар як соат тӯли 3 дақиқа ранги сабз равшанӣ медиҳад, пас аз он тӯли як сония ранги зард ва баъд тӯли ду сония ранги сурх, тӯли се сония боз ранги сабз ва ғ. (ва боз такрор меёбад). Қимати ҳақиқии t , ки вақтро бо дақиқа аз оғози вақти гузашта ифода мекунад дода шудааст. Муайян карда шавад, ки дар ин лаҳзаи вақт кадом ранг равшанӣ медиҳад.
18. Робот метавонад, ба чор самт ҳаракат карда, («Ш» — шимол, «Ҷ» — ҷануб, «Ғ» — ғарб, «С» — шарқ) 3 фармони рақамӣ қабул кунад: 0 — давом додани ҳаракат, 1 — гардиш ба чап, -1 — гардиш ба рост. Рамзи s — мавқеи ҷорӣи робот ва адади бутуни N — фармони ба он равонкардашуда дода шудааст. Самти роботро пас аз қабул кардани ин фармон муайян кунед.
19. Бо дохил кардани ададҳои дурақамаи аз 20 то 99 номи рақамро хориҷ кунед.
20. Бо дохил кардани ададҳои аз 1 то 1000 номи рақамро хориҷ кунед.

Масъалаҳои гурӯҳи For

- 1) Ададиҳои бутуни n ва k дода шудаанд. n маротиба адади k -ро хориҷ кунед.
- 2) Суммаи ҳамаи ададҳои мукаммали порчаи $[2..5000]$ -ро ҳисоб кунед.
- 3) Суммаи қисмҳои даҳии ҳамаи ададҳои серақамаро ҳисоб кунед.
- 4) Ададҳои содаи порчаи аз 1 то 1000 ёфта шаванд.
- 5) Суммаи $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$ ҳисоб карда шавад.
- 6) Суммаи $\frac{2}{3} + \frac{3}{4} + \frac{4}{5} + \dots + \frac{10}{11}$ ҳисоб карда шавад.
- 7) Суммаи $1 + \frac{1}{3} + \frac{1}{3^2} + \dots + \frac{1}{3^8}$ ҳисоб карда шавад.
- 8) Суммаи $1 - \frac{1}{2} + \frac{1}{3} - \dots + (-1)^{n+1} \frac{1}{n}$ ҳисоб карда шавад. Оператори шартӣ истифода бурда нашавад.
- 9) Барномае нависед, ки бо дода шудани x қимати $\cos(x)$ -ро, ки ба намуди қатор дода шудааст ҳисоб кунад.

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!}$$

- 10) Адади бутуни N ($N > 0$) дода шудааст. Суммаи зеринро ҳисоб кунед. $1^N + 2^{N-1} + \dots + N^1$
- 11) Маблағи дар бонк гузошташуда 1000 сомонӣ аст. Ҳар моҳ ҳаҷми он P фоиз зиёд мешавад (P -адади ҳақиқӣ ($0 < P < 25$)). Бо фоизи додашудаи P муайян кунед, ки пас аз чанд моҳ маблағи гузошташуда аз 2200 с. зиёд мешавад. Микдори ёфташудаи моҳҳо K (адади бутун) ва ҳаҷми умумии маблағи дар бонк гузошташуда пас аз зиёдшавӣ S (S -адади ҳақиқӣ)-ро хориҷ кунед.
- 12) Барномае нависед, ки маълумотҳои зеринро дар чунин шакл хориҷ кунад:

a)	б)
5	1 1 1 1 1
5 5	1 1 1 1
5 5 5	1 1 1
5 5 5 5	1 1
	1

- 13) Барномае нависед, ки дар намуди чадвали зерин ададҳоро хориҷ кунад:

a)	б)
1	10
2 2	20 20
3 3 3	30 30 30
4 4 4 4	40 40 40 40

14) Барномае нависед, ки дар намуди чадвали зерин ададхоро хориҷ кунад:

а)					б)				
0					20	21	22	23	24
1	0				19	20	21	22	
2	1	0			18	19	20		
3	2	1	0		17	18			
4	3	2	1	0					

15) Барномае нависед, ки чадвали зерин хориҷ кунад:

$$\begin{array}{lll}
 1+1=2 & 2+1=3 & \dots \quad 9+1=10 \\
 1+2=3 & 2+2=4 & \dots \quad 9+2=11 \\
 \dots & \dots & \dots \quad \dots \\
 1+9=10 & 2+9=11 & \dots \quad 9+9=18
 \end{array}$$

16) Барномае нависед, ки чадвали зерин хориҷ кунад:

$$\begin{array}{lll}
 1 \times 1 = 2 & 1 \times 2 = 3 & \dots \quad 1 \times 9 = 10 \\
 2 \times 1 = 3 & 2 \times 2 = 4 & \dots \quad 2 \times 9 = 11 \\
 \dots & \dots & \dots \quad \dots \\
 9 \times 1 = 10 & 9 \times 2 = 11 & \dots \quad 9 \times 9 = 18
 \end{array}$$

17) Адади натуралии n ($n \leq 9$) дода шудааст. Талаб карда мешавад, ки барномаи ҳисобкунии қимати ифодаи $1 + 111 + 11111 + \underbrace{11 \dots \dots 11}_{n\text{-маротиба}}$ барнома

тартиб дода шавад.

18) Адади натуралии n ($n \leq 9$) дода шудааст. Талаб карда мешавад, ки барномаи ҳисобкунии қимати ифодаи $1+121+12321+12\dots n\dots 21$ барнома тартиб дода шавад.

19) Адади натуралии ($1 \leq N \leq 10^4$) дода шудааст. N –то ададҳои аввале, ки аз тақсимкунандаҳои сода танҳо 2, 3, 5-ро доранд (пайдарпайии Хемминг), бо тартиби афзуншавиашон ба чоп бароред. Масалан 10 адади аввалае, ки ин шартро қаноат мекунонд, инҳоянд: 2, 3, 4, 5, 6, 8, 9, 10, 12, 15

20) Барномае нависед, ки қимати суммаи зеринро ҳисоб кунад.

$$\prod_{i=1}^n \sum_{j=1}^m \prod_{k=1}^t (i^3 + j^2 + k)$$

21) Адади натуралии x дода шудааст. Миқдори тарзҳои дар суммаи чор адади натуралӣ тасвир кардани он: $x = a + b + c + d$ -ро ёбед, ки дар ин ҷо $a \leq b \leq c \leq d$ мебошанд

22) Адади натуралии n ($n \leq 9$) дода шудааст. Талаб карда мешавад, ки барномаи ҳисобкунии қимати ифодаи $1+22+333+n\dots n$ барнома тартиб дода шавад.

23) Ададҳои аз 1 то n ($2 \leq n \leq 10^6$) –ро бо миқдори имконпазири максималии чуфтҳо тавре чудо кардан лозим аст, ки суммаи ададҳо дар чуфтҳо адади сода

бошад. Миқдори чуфтҳоро чоп кунед. Масалан, ҳангоми $n=3$ як чуфт (1,2) ва ҳангоми $n=7$ сеои ин гуна чуфт-(1,2), (3, 4) ва (6, 7) мешаванд.

24) Пайдарпайии ададҳои бутун 21, 12, 11, ... дода шудааст. Чор аъзои навбатии ин пайдарпайиро муайян кунед.

Масъалаҳои гурӯҳи While

- 1) Адади натуралӣ дода шудааст. Муайян карда шавад, ки оё пайдарпайии рақамҳои он аз тарафи рост ба чап ба таври афзуншавӣ ҷойгир шудаанд ё не. Масалан, барои адади 5321 ҷавоб дуруст аст, барои адади 5213 бошад ҷавоб нодуруст аст.
- 2) Адади ҳақиқии a дода шудааст. Қимати n , ки барояш $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} > a$ аст, ёфта шавад.
- 3) Адади бутуни мусбат дода шудааст, муайян кунед, ки он ба 7 каратӣ мебошад?
- 4) Ду адади бутуни мусбати A ва B ($B > A$) дода шудааст. Амалҳои тақсирро истифода набурда муайян кунед, ки натиҷаи тақсими B бар A чанд мешавад?
- 5) Шаҳрванд дар бонк барои худ суратхисоб кушода дар он 1000 сомонӣ гузошт. Ҳар моҳ маблағи гузошташуда 2% аз ҳисоби маблағи мавҷудбуда зиёд мешавад. Муайян карда шавад, ки суммаи маблағи гузошташуда пас аз 3, 4, ..., 12 моҳ чанд сомони ро ташкил медиҳад.
- 6) Суммаи ҳамаи ададҳои дурақамаро ёбед, ки решаи квадратии онҳо адади тоқ мебошад.
- 7) Қори Исмаи як миқдор тангаҳои тиллоиро дар сандуқ ҷойгир намуда, соли дуум аз он ҷо якчанд тангари гирифт. Аз соли сеум сар карда, вай ба сандуқ ба миқдори тангаҳои ду соли пешина танга ҳамроҳ мекард.
Талаб карда мешавад, ки барномаи муайянкунии миқдори тангаҳоеро, ки дар соли дуум аз сандуқ гирифтани лозим аст, сохта шавад, агар дар соли якум ба сандуқ Y -то танга ҷойгир намуда, дар соли X -ум дар сандуқ Z -то танга мавҷуд бошад.
- 8) Маблағи дар бонк гузошташуда 1000 сомонӣ аст. Ҳар моҳ ҳаҷми он P фоиз зиёд мешавад (P -адади ҳақиқӣ ($0 < P < 25$)). Бо фоизи додашудаи P муайян кунед, ки пас аз чанд моҳ маблағи гузошташуда аз 2200 зиёд мешавад. Миқдори ёфташудаи моҳҳо K (адади бутун) ва ҳаҷми умумии маблағи дар бонк гузошташуда пас аз зиёдшавӣ S (S -адади ҳақиқӣ)-ро хориҷ кунед.
- 9) Адади бутуни a дода шудааст. Муайян карда шавад, ки он адади дараҷаи 9 аст.
- 10) Адади бутуни a дода шудааст. Муайян карда шавад, ки он адади дараҷаи 3 аст.
- 11) Адади бутуни a дода шудааст. Муайян карда шавад, ки дар навишти он адади 3 воমেхурад ё не?
- 12) Адади бутуни N (> 0) дода шудааст. Чунин адади хурдтарини K -ро ёбед ки квадратии он аз N зиёд шавад: $K^2 > N$. Функцияи аз решабарорӣ истифода бурда нашлавад.

13) Адади бутуни $N (> 0)$ дода шудааст. Чунин адади калонтарини K -ро ёбед ки квадрати он аз N зиёд нашавад : $K^2 \leq N$. Функцияи аз решабарорӣ истифода бурда нашавад.

14) Адади бутун дода шудааст. Ададери хориҷ кунед, ки аз рақамҳои он вале бо тартиби баръакс ҷойгир бошанд.

15) Адади бутуни $N (> 1)$ дода шудааст. Чунин адади хурдтарини K -ро ёбед ки барои он нобаробарии $3^K > N$ иҷро шавад.

16) Адади бутуни $N (> 1)$ дода шудааст. Чунин адади калонтарини K -ро ёбед ки барои он нобаробарии $3^K < N$ иҷро шавад.

17) Адади ҳақиқии $\varepsilon (\varepsilon > 0)$ дода шудааст. Пайдарпайии ададҳои ҳақиқии A_K чунин муайян кра мешаванд:

$$A_1 = 2, \quad A_K = 2 + 1/A_{K-1}, \quad K = 2, 3, \dots$$

Рақами тартибии нахустини K ёфта шавад, ки барои он шарти зерин иҷро мегардад

$$|A_K - A_{K-1}| < \varepsilon,$$

аъзоҳои A_{K-1} ва A_K хориҷ карда шаванд.

18) Адади ҳақиқии $\varepsilon (\varepsilon > 0)$ дода шудааст. Пайдарпайии ададҳои ҳақиқии A_K чунин муайян карда мешаванд:

$$A_1 = 1, \quad A_2 = 2, \quad A_K = (A_{K-2} + 2 \cdot A_{K-1})/3, \quad K = 3, 4, \dots$$

Рақами тартибии нахустини K ёфта шавад, ки барои он шарти зерин иҷро мегардад

$$|A_K - A_{K-1}| < \varepsilon,$$

аъзоҳои A_{K-1} ва A_K хориҷ карда шаванд.

19) Факториали адад дода шудааст. Худи адад муайян карда шавад.

20) Хурдтарин қаратноки умумии ду адади натуралии додашуда ёфта шавад.

21) Адади натуралий дода шудааст. Миқдори рақамҳои он ва суммаи рақамҳои он ҳисоб карда шавад.

Боби 5. Функцияҳо(functions)

Муқаддима

Дар мавзӯҳои қаблӣ мо ба функцияҳои математикӣ ва ё стандартии забони C++ шинос шудем ва аз онҳо ҳам истифода кардем. Тавре ки дидем, ин функцияҳо дар китобхонаи `cmath` (`math.h`) ҷойгир буда, ҳангоми даъват дар диҳох ҷойи барнома ҳозир шуда, амали хоссеро анҷом медиҳанд. Аксари забонҳои барномасозӣ имконият медиҳанд, ки барномаҳои ҳаҷман калонро ба якҷанд қисматҳои ба ҳам вобаста ҷудо намуда, ҳал кунем ва дар охир онҳоро якҷоя намуда, ҳалли матлубро ба даст оварем. Таҷрибаҳо нишон доданд, ки агар барномаи калонро ба қисматҳои хурд ба номи «unit» ва ё «modul» ҷудо кунем, идора намудани онҳо хеле сода мешавад. Дар охир ин қисматҳоро гирд оварда, дар дохили барномаи асосӣ истифода мекарем. Масалан, талаб карда шавад, ки қимати ифодаҳои зерин ҳисоб карда шавад.

$$1) \quad Z = \begin{cases} \min(e^x, \pi, c) + \sum_{i=1}^n (P_i! - x!) & x > 0 \\ \max(a^x, c, \pi) + \sum_{j=1}^m P_j!! + a! - \max(a, c, c) & x \leq 0 \end{cases}$$

$$2) \quad S = 5! + 8! + 12! + 9! - C_7^{11}$$

Ба намуди умумӣ барои ҳал намудани ин ифодаҳо, ба барноманавис лозим аст, ки барои якҷанд қисматҳоро такроран код нависад. Масалан, барои ҳал намудани мисоли ду бояд барои ҳар як қимат алоҳида факториалро ҳисоб намуда, сипас онҳоро якҷоя намуда ҳалли матлубро ба даст орад. (Ин мисолҳо танҳо барои намуна оварда шудаанд, ҳол онки метавон мисолҳои дигар низ овард ва ҳар як мисолро бо роҳ ва усулҳои гуногун ҳал кард).

Барои ба таври содатар ҳал намудани ин ифодаҳо оҳоро ба қисматҳо ҷудо намуда, барои ҳар қисмати барномаи алоҳида менависем ва дар охир онҳоро ҷамъ намуд, ҳалли лозимро меёбем. Барои фаҳмо шудан мисоли дуввумро бо ду роҳ ҳал мекунем, то хонандаи азизи мо ба ин матлаб беҳтар сарфаҳм равад.

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
    long int S1 = 1, S2 = 1, S3 = 1, S4 = 1, S5 = 1, S6 = 1, S7 = 1;
    float S;
    for (int i = 1; i <= 5; i++)
        S1 = S1*i;
    for (int i = 1; i <= 8; i++)
        S2 = S2*i;
    for (int i = 1; i <= 12; i++)
        S3 = S3*i;
    for (int i = 1; i <= 9; i++)
        S4 = S4*i;
    for (int i = 1; i <= 7; i++)
```

```

        S5 = S5*i;
    for (int i = 1; i <= 1; i++)
        S6 = S6*i;
    for (int i = 1; i <= (11 - 7); i++)
        S7 = S7*i;
    S = S1 + S2 - S3 + S4 - (S5) / (S6*S7);
    cout << S << "\n";
    return 0;
}

```

Тавре ки дида мешавад, дар ин чо аз 8 тағйирёбанда (ба ғайр аз i) ва 7 давр (сикл) истифода шудааст. Чи таре ки медонем, истифодаи тағйирёбандаи зиёд аз як тараф хотираи зиёдро банд намуда, қобилияти иҷроиши барномаро суст мегардонанд. Аз тарафи дигар бошад, чанд маротиба такроран навиштани як қитъаи барнома хело дилгиркунанда буда, матни барномаро зиёд мекунад ва ба вучудоии хатогиҳоро сабаб мешавад. Агар монанди функсияҳои стандартӣ дар китобхонаи C++ як функсия барои ҳисоб намудани факториал вучуд меошад матни барнома ин қадар калон намешуд. Чунин функсияҳоро дар забони C++ худи барномасоз метавонад низ эҷод кунад. Ҳоло мисоли болоро бо истифода аз функсияи навиштаи барномасоз ҳал мекунем.

```

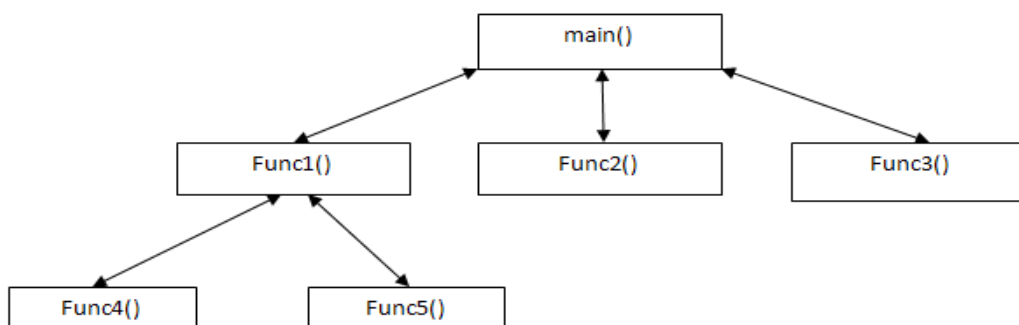
#include <iostream>
using namespace std;
long int fact(int n){
    long int P = 1;
    for (int i = 1; i <= n; i++)
        P = P*i;
    return P;
}
int main(){
    float S;
    S = fact(5) + fact(8) - fact(12) + fact(9) -
        fact(11) / (fact(7)*fact(11 - 7));
    cout << S << "\n";
    return 0;
}

```

Тавре ки дида мешавад, дар ин барнома ягон қисмат такрор нашуда, тағйирёбандаҳои иловагӣ низ вучуд надоранд. Амалкарди барномаро баъдтар ширх медиҳем.

Ҳангоми ба қисматҳо ҷудо намудани барнома, ҳар қисматро зербарнома (sub program) мегӯянд. Зербарнома дар ҳар забонҳои барноманависӣ ба таври гуногун буда, номҳои махсусро дорад. Маса-лан, дар забонҳои Паскал зербарномаро модул ё процедуре ва дар забонҳои Бейсик function мегӯянд. Дар забони C++ бошад, ин қисматҳоро функсия (тобеъ) мегӯянд. Пас, барои ҳалли масъалаи калон онро ба қисматҳо ҷудо намуда, барои ҳар қисмат барномаи алоҳида менависем ва дар охир ҳамаи қисматҳоро дар як функсияи асосӣ якҷо ҷамъ мекунем. Ин равишро барномасозии модульӣ (modular programming) мегӯянд.

Дар забони C++ чунон ки қаблан ишорат кардем, як функсияи асли ба номи main () вуҷуд дорад, ки барнома кори худро аз он оғоз мекунад. Инчунин, тамоми функсияҳои фарӣ дар он гирд оварда мешаванд. Дар забони C++ ду намуди функсияҳо мавҷуданд: якум, функсияҳои стандартӣ ва ё китобхонаӣ (library functions), ки онҳо функсияҳои тайёр (build in functions) низ мегӯянд, ки бо пайваст кардани китобхонаи лозимӣ ба кор омода мегарданд, ба монанди функсияҳои математикӣ, дохилкунию хориҷкунӣ ва ғайра. Дуюм, функсияҳо, ки худ барномасоз барои осон намудани идоракунии барнома менависад ва дар ҷои лозимии барнома аз онҳо истифода мекунад. Ин намуд функсияҳоро функсияҳои фарӣ меноманд. Ба намуди диаграмма силсилаи функсияҳо намуди зеринро доранд.



Шояд баъзе хонандагони мо суол кунанд, ки ба таври мушаххас функсия барои чӣ лозим аст ва он барномасоз чӣ имкониятро медиҳад?

Барои ба ин савол ҷавоб гирифтани матлаби зеринро дида мебароем, ки иборат аст аз чунин имкониятҳои функсия:

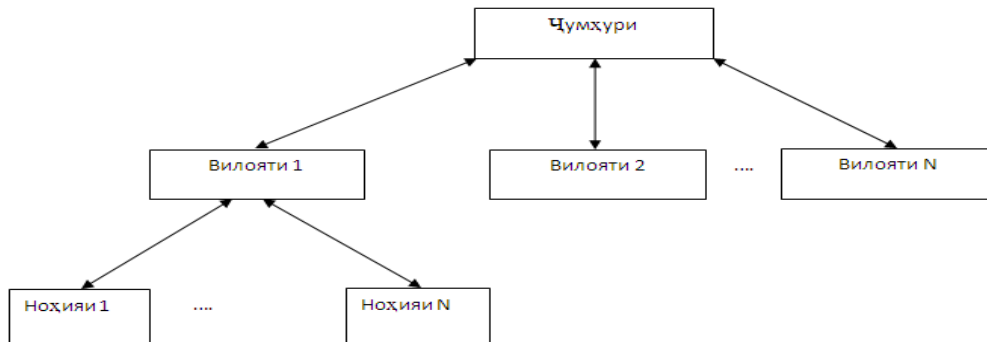
1. Осон шудани идоракунии барнома - вақте ки як барномаи ҳаҷман бузург ба якчанд қисматҳо тақсим карда шуд. Ҳар як қисмат алоҳида буда, амали хоссеро иҷро мекунад, аз ин рӯ, метавонем ҳар яки онҳоро ба таври алоҳида навишт ва ҳар кадоми онҳоро санҷиш намуд. Агар онҳо беҳато бошанд, пас, дар функсияи асосӣ онҳоро ба кор мебарем. Ин амал ислоҳи хатогиҳо ва тағйири барномаро хеле осон мегардонад.

2. Ба таври осон ба барнома тағйирот ворид кардан - агар зарурате бошад, ки ба як қисмати барнома тағйироте дохил карда шавад, зарур нест, ки тамоми барнома аз назар гузаронида таҳлил кард, балки тағйироти зарурӣ вобаста ба функсияе, ки ба он тағйирот алоқаманд аст дохил карда мешавад.

3. Қобилияти истифодаи аз сари нав - баъзан зарурати истифодаи як қисмати барнома дар якчанд ҷойҳо пеш меояд. Агар он қисматро ба намуди як функсия нависем, дигар зарурате намешавад, ки он қисматро пурра дар ҷойи лозимӣ аз сари нав навишта шавад, танҳо даъват намудани он кифоя аст. Ин амал коди барномаро хело сода ва кӯтоҳ мегардонад.

Барои ба моҳияти функсия сарфаҳм рафтан мисолеро дида мебароем: ҳамаи мо медонем, ки ҳар давлат дар худ шох ҳуқумрон дорад. Худи шох наметавонед тамоми қорҳои давлатиро иҷро намояд ва ҳамаи қисматҳои давлатро идора кунад, аз ин ҷо зарурат ба шахсоне пайдо мешаванд, ки бо қорҳои давлатӣ ва идоракунии ба ӯ кӯмак кунанд. Аз ин рӯ, давлатро ба вилоятҳо, шаҳрҳо, ноҳияҳо, ҷамоаҳо ва деҳаҳо тақсим мекунад ва барои ҳар

кадомаш як роҳбаре таъйин мекунад. Роҳбарони мазкур бар замми он ки дар ягон ҷо роҳбаранд, ба роҳбари олий, яъне шоҳ итоат мекунанд. Бо ин равиш идоракунии як давлати бузург хеле сода мегардад. Функцияҳо дар забони C++ низ ба ин сурат амал мекунанд. Тавре ки қаблан гуфтем, функцияи асосӣ (main) ҳамчун шоҳ хизмат намуда, функцияҳои боқимонда ба он пайравӣ мекунанд ва ҳамчуноне, ки як вилоят ба ноҳияҳо тақсим мешавад ва ё дар дохилаш якчанд вилоятҳоро дар бар мегирад. Як функция ҳам метавонад дар дохилаш функцияи дигарро нигоҳ дорад, яъне барои функцияи дигар функцияи аслии бошад. Мисоли овардаамонро метавонем тавассути нақшаи зерин тасвир кунем.



Тарзи муаррифӣ (declaration) намудани функцияҳо

Тарзи муаррифӣ намудани як функцияро дар мавзӯи гузашта баён карда будем. Ҳоло бошад, дар ин мавзӯъ онро ба таври пурратар дида мебароем. Ба намуди умумӣ тарзи муаррификунии як функция шакли зеринро дорад:

```

data_type func_name(data_type argv1, data_type argv2,...)
{
    title function;
    return value
}
    
```

Ё ба таври дигар:

```

<навъи функция> <номи функция > <аргументҳои функция>
Муаррифӣ намудани тағйирёбандаҳо
{
    Муаррифӣ намудани тағйирёбандаҳои маҳаллӣ
    Тани функция (яъне маҷмӯи операторҳо)
    Қимат баргардонидани функция
}
    
```

Дар ин ҷо func_name-номи функция (яъне ҳар функцияе, ки мо менависем, дорои ном аст, ки он ном дар функцияи аслии даъват карда мешавад.), data_type-навъи функция (функция бояд муайян бошад, ки чӣ гуна қимат бармегардонад, яъне қимати баргарданда аз кадом навъи стандартӣ аст), ва argv1, argv2,...-аргументҳои функция мебошанд (дар сурати вучуд доштани аргумент.), title func-тан ва ё амали иҷро мекардаи функция, қавсҳои фигуравӣ ({}), маънои ибтидо ва интиҳо (begin ва end)-и функция (ҳар функция ва барнома дар дилхоҳ забони барномасозӣ бо қалима ва ё мафҳуме оғоз гардида бо мафҳуме ба охир мерасад) ва return value бошад, қимате, ки функция бармегардонад мебошад.

Тавре ки мушоҳида кардем, як функция дорои ном ва навъ буда, дар дохили қавс аргументҳо ва навъи онҳо навишта мешаванд, албатта, дар сурати вучуд доштани аргументҳо. Яъне бояд функция пурра муайян бошад, ки чӣ гуна аргумент қабул мекунад ва чӣ гуна қимат бармегардонад.

Дар функция ба ғайр аз аргументҳо, Тавре ки пештар гуфтем, тағйирёбандаҳои дохили қавси баъди номи функция омада, инчунин тағйирёбандаҳои маҳаллӣ (local variables declaration) метавонанд муаррифӣ ва истифода шаванд.

Тағйирёбанда агар чӯзӣ аргументҳои функция набошад, яъне худ функция бошад, маҳаллӣ номида мешавад. Тағйирёбандаҳои маҳаллӣ пас аз унвони функция навишта мешаванд.

Унвони функция - ин номи функция, навъи функция, аргументҳои функция ва навъи аргументҳои функция мебошад.

Оператори return

Барои баромадан аз функция дар вақти ба интиҳои функция расидан оператори return истифода бурда мешавад. Агар лозим шавад, ки функция ягон қимат баргардонад, оператори мазкур дорои аргумент мешавад. Ба намуди пешфарз аргументи ин оператор 0 (сифр) мебошад. Ин оператор аргументашро аввал ба номи функция ихтисос медиҳад ва баъд идоракунӣ ба функцияи дигар мегузарад.

Мисол. Функцияе нависед, ки квадрати ададҳои натуралии фосилаи аз 1 то 10 –ро ҳисоб намуда, ба функцияи асли баргардонад

<pre>#include <iostream> using namespace std; int Kvadrat(int n) { return n*n; } int main() { for (int i = 1; i <= 10; i++) cout << i << " ^ 2 = " << Kvadrat(i) << " \n "; return 0; }</pre>	<pre>// номи функция, навъи функция , аргументи функция ва навъи аргументи функция // оператори return қимати n*n -ро бармегардонад.</pre>
--	--

Тавре ки мушоҳида гардид, ин функция қимати i -ро аз функцияи асли (main) ба параметри n нусхабардорӣ намуда, баъд бо ёрии оператори return $n*n$ қимати онро ба квадрат бардошта ба функцияи асли бармегардонад.

Маҳдудияти оператори return дар он аст, ки он фақат як миқдорро ба функция бармегардонад.

Навъи void навъест ягон қимат барнамегардонад. Вақте ки функция бо ин навъ навишта мешавад, зарурати истифодаи оператори return ба миён намеояд. Масалан, мисоли пешина ҳангоми навъи функция void-будан дар шакли зерин навишта мешавад.

```

#include <iostream>
#include <cmath>
using namespace std;
void Kvadrat(int n){
    for (int i = 1; i <= n; i++)
        cout << i << " ^ 2 = " << i*i << " \n ";
}
int main() {
    Kvadrat(10);    //даъвати функция
    return 0;
}

```

Функцияи мазкур қимате барнамегардонад, зеро он аз навъи холӣ (void) муаррифӣ шудааст ва дар фурухонии функция номи он фақат дар як оператор истифода шудааст. Натиҷаи ин мисол низ монанди мисоли боло мебошад. Тавре ки аз ин мисолҳо мушоҳида шуд, номи функция фақат дар оператори чопкунӣ навишта шуд, яъне функция дар вақти чоп намудан даъват карда шуд, илова бар ин, метавон номи функцияро дар ягон ифодаи ҳисобкунӣ истифода кард. **Мисол:**

```

#include <iostream>
#include <cmath>
using namespace std;
int Kvadrat(int n) {
    return n*n;
}
int main(){
    double LOG;
    for (int i = 1; i <= 10; i++) {
        LOG = log((double)Kvadrat(i)); // Истифодаи функция
        cout << "log(" << i << ") = " << LOG << " \n ";
    }
    return 0;
}

```

Оператори return на танҳо дар интиҳои барнома, инчунин метавонад дар дилхоҳ ҷойи барнома, ки зарурат бошад, истифода шавад.

Мисол: Қимати ифодаи зеринро бо истифода аз функция ҳисоб кунед.

$$F(x) = \begin{cases} -2x + 8, & x < 0 \\ 2x - 5, & x \geq 0 \end{cases}$$

Матни барнома.

```

#include <iostream>
using namespace std;
double F(double x){
    if (x < 0)
        return -2 * x + 8;
    else
        return 2 * x - 5;
}
int main(){
    double x, i = 0;
}

```

```

while (++i <= 2) {
    cout << "x = "; cin >> x;
    cout << "F(" << x << ") = " << F(x) << " \n ";
}
return 0;
}

```

Тавре ки дидем, дар ин барнома оператори return дар функция ду маротиба истифода шуд, ба ҳамин шаклаш метавонад дар дилхоҳ ҷойи функция истифода шавад. Дар функцияи асли истифодаи дастури даврии бо шарти пешоянд саҳифаи тасвириро то дохил кардани ду рақам нигоҳ медорад.

Аз мисолҳои фавқ маълум гардид, ки дар функцияи фаръӣ тағйирёбандаҳое, ки ба сифати аргумент истифода шуданд, аз тағйирёбандаҳои функцияи асли истифодашуда фарқ мекунанд. Инчунин онҳо метавонанд ҳамном бошанд. Монанди мисоли дидабаромадамон ҳам дар функцияи фаръӣ ва ҳам дар функцияи асли аргумент тағйирёбандаи x истифода шудааст. Тағйирёбандаҳои функцияи фаръӣ тағйирёбандаҳои маҷозӣ (dummy variables) номида мешаванд. Барои онҳо дар хотира катакча (ҷое) ҷудо карда мешавад.

Ба функция раво кардани параметрҳо

Аз функцияи асли ба функцияи фаръӣ аргументҳо бо ду тарз интиқол дода мешаванд:

- 1) Интиқол аз рӯи қимат (pass-by value)
- 2) Интиқол аз рӯи истинод (pass-by reference)

Дар равиши аввал худ аргументҳо ба функцияи фаръӣ фиристода намешаванд, танҳо нусха (copy)-и онҳо фиристода мешавад. Аз ин сабаб ҳар тағйироте, ки дар функцияи фаръӣ ба амал меояд, ҳеҷ таъсире ба функцияи асли намегузорад. **Мисол:**

```

#include <iostream>
using namespace std;
int davatazruiqim(int adad){
    adad += 5;
    return adad;
}
int main(){
    int N;
    cout << "N = "; cin >> N;
    cout << "Istifodai tobei davatazruiqim = " << davatazruiqim(N) << endl;
    cout << "N = " << N << endl;
    return 0;
}

```

Дар мисоли мазкур пас аз даъвати функция, қимати тағйирёбандаи N тағйир наёфт, балки ҳамон тавре, ки буд, ҷой гардид. Натиҷаи барнома ба сурати зерин аст.

```

N=18
Istifodai tobei davatazruiqim=23
N=18

```

Дар равиши дуҷум бошад, суроғаи аргументҳо дар функсияи аслии нишон дода мешавад, аз ин рӯ, функсия метавонад бе нусхабардории аргумент бо он дастрас шавад ва ҳар амале, ки дар функсияи фаръӣ болои аргумент ба амал меояд, он дар функсияи аслии низ болои он аргумент таъсир мерасонад. Барои равоӣ кардани истиноди параметрҳо ба функсия аз оператори амперсанд (&) истифода бурда мешавад. **Мисол:**

```
#include <iostream>
using namespace std;
int davatazruiqim(int &N){
    N += 5;
    return N;
}
int main(){
    int N;
    cout << "N = "; cin >> N;
    cout << "Istifodai tobei davatazruiqim = " << davatazruiqim(N) <<
endl;
    cout << "N = " << N << endl;
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
N=18
Istifodai tobei davatazruiqim=23
N=23
```

Дар ин равиш тавре ки аз натиҷа бармеояд, тағйиротро ки дар функсияи фаръӣ болои аргумент ба амал омад, он тағйирот ба аргумент дар функсияи фаръӣ асар гузошт. (Минбаъд дар ин бора эзоҳи пурратар дода мешавад.)

Мисол. Барномае нависед, ки факториали дукаратаи адади натуралии n -ро ҳисоб кунад.

Қайд: *Факториали дукарата дорои ду махсусияти зерин аст:*

{ $N!! = 1 * 3 * 5 * 7 * \dots * N$, агар N тоқ бошад;
 { $N!! = 2 * 4 * 6 * 8 * \dots * N$, агар N чуфт бошад;

```
#include <iostream>
using namespace std;
double Fact2(double n){
    double f = 1, k;
    k = ((int)n % 2 == 0) ? 2 : 1;
    for (int i = k; i <= n; i += 2) f = f*i;
    return f;
}
int main()
{
    int N;
    cout << "N = "; cin >> N;
    cout << N << "!! = " << Fact2(N) << " \n ";
    return 0;
}
```

Функсияҳои бозгаштӣ (Рекурсивний функция)

Функсияҳои бозгаштӣ функсияҳои мебошанд, ки дар онҳо аз дохили функсия боз худӣ функсия чег зада мешавад. Барои ба ин матлаб пурратар шинос шудан биёед факториали ададро дида мебароем. Факториали адади бутуни n чунин муайян карда мешавад:

$$n! = \begin{cases} 1 & \text{агар } n \leq 1 \\ n(n-1)! & \text{дар дигар ҳолатҳо} \end{cases}$$

Ба мо маълум аст, ки факториали адад ҳосили зарби ададҳои аз як то худӣ адад мебошад, яъне

$$n! = 1 * 2 * 3 * \dots * (n-1) * n$$

аз ин ҷо $n! = n * (n-1)!$

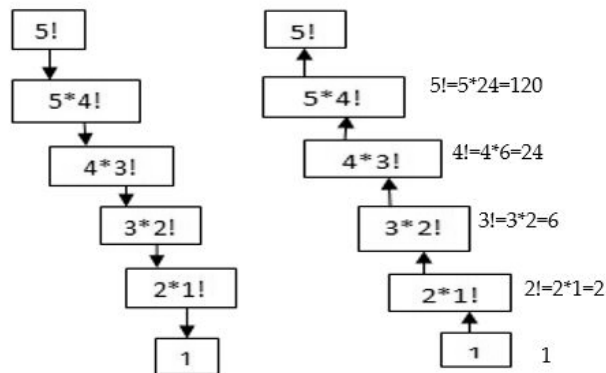
Тавре ки дидем, дар сатри дуввум факториал мустақиман ҳисоб нашуда, балки ба намуди $n(n-1)!$ таъриф шудааст.

Дар функсияҳои бозгаштӣ ин амал чунин иҷро карда мешавад:

$$n! = n * (n-1) * (n-2) * \dots$$

Ин амал то замоне давом мекунад, ки аргументи функсия ба 1 баробар шавад, яъне: $n! = n * (n-1) * (n-2) * \dots * 1$

дар охир сар карда, ҳамаи элементҳоро ба ҳамдигар зарб мекунад:



Қараёни даъвати бозгашт (рекурсия). Микдоре, ки аз ҳар даъват бармегардад.

Мисол. Адади бутуни n дода шудааст, факториалӣ онро бо истифода аз функсияи бозгаштӣ ҳисоб кунед.

```
#include <iostream>
using namespace std;
unsigned long int factorial(unsigned int n){
    if (n <= 1)
        return 1;
    else
        return n*factorial(n - 1);
}
int main() {
    unsigned int n;
    cout << "n = "; cin >> n;
    cout << n << " != " << factorial(n) << endl;
    return 0;
}
```

Дар забонҳои Basic ва Fortran 77 равиши бозгаштӣ пешбинӣнашудааст. Дар баъзе забонҳои дигар ин равиш пешбинӣ шудааст, вале ҳангоми муаррифӣ намудани функсия бояд ба таври сареъ муайян карда шавад, ки функсия аз навъи бозгаштӣ аст ё не, то компилятор онро аз дигар функсияҳо фарқ кунад.

Барои ба моҳияти ин навъ функсияҳо шинос шудан мисоли зеринро дида мебароем.

Мисол. Суммаи n ададӣ аввали натуралӣро ҳисоб кунад.

```
#include <iostream>
using namespace std;
long int Summa(unsigned int n){
    if (n <= 1)
        return n;
    else
        return n + Summa(n - 1);
}
int main(){
    unsigned int n;
    cout << "n = "; cin >> n;
    cout << "Summai " << n << " adadi avalai naturali\nSumma = " <<
Summa(n) << endl;
    return 0;
}
```

Пешнамунасоӣ аз функсия (Function Prototypes)

Дилхоҳ функсия дар забони C++ бояд то замони даъват муаррифӣ карда шавад. Ин ҳел тавсифномаи пешнамунасоии функсия меноманд. Наҳви пешнамунасоии функсия ба сурати зерин аст:

```
data_type func_name(data_type 1, data_type 2,...)
```

Ин навъ имконият медиҳад, ки дар сурати яксон (якнавъ) набудани аргументҳо ва ё дар сурати баробар набудани миқдори аргументҳо аз вучуд омадани хатогиҳо пешгирӣ кунад.

Дилхоҳ функсияе, ки пеш аз функсияи асосӣ муаррифӣ мегардад, ҳамчун пешнамуна фаҳмида мешавад.

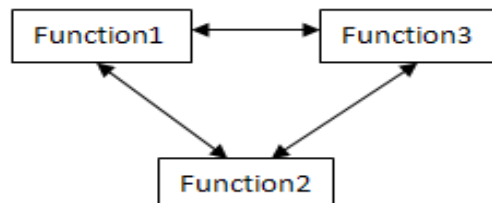
Мисол. Функсияе нависед, ки суммаи ададҳои токи натуралии аз 1 то 10-ро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int SummaAtoq(int n);
int main(){
    cout << "SummaAtoq = " << SummaAtoq(10) << endl;
    return 0;
}
int SummaAtoq(int n){
    int S = 0;
    for (int i = 1; i <= n; i += 2)
        S += i;
    return S;
}
```

Натиҷаи барнома: SummaAToq=257

Истифодаи якчанд функсия дар як барнома

Дар забони C++ ба ҳар микдоре, ки лозим бошад, метавон функсия навишта (муаррифӣ кард) истифода кард. Ҳар функсия метавонад функсияи дигарро даъват кунад. Дар забони Паскал ва дигар забонҳо функсияе, ки дар дохили як функсия таъриф шудааст, наметавонад бо ёрии функсияи дигар ҷеғ зада шавад. Дар забони C++ ҳамаи функсияҳо баробар буда, ҳуқуқҳои баробар доранд ва ҳар функсия метавонад функсияи дигарро фурӯзонӣ (даъват) кунад. Робитаи байни функсияҳо дар забони C++ ба сурати зерин аст:



Тавре ки аз ин расм бармеояд, ҳар функсия имконият дорад, ки бо функсияи дигар мустақиман ва ё ғайримустақиман робита дошта бошад.

Мисол. Барномае нависед, ки дар он аз якчанд функсия истифода карда шавад.

```
#include <iostream>
using namespace std;
double Kvadrat(double x){
    return x*x;
}
double Cube(double x) {
    return Kvadrat(x)*x;
}
double DarChor(double x) {
    return Kvadrat(x)*Kvadrat(x);
}
double DarPanj(double x) {
    return Kvadrat(x)*Cube(x);
}
int main(){
    double x;
    cout << "x = "; cin >> x;
    cout << x << " ^ 2 = " << Kvadrat(x) << endl;
    cout << x << " ^ 3 = " << Cube(x) << endl;
    cout << x << " ^ 4 = " << DarChor(x) << endl;
    cout << x << " ^ 5 = " << DarPanj(x) << endl;
    return 0;
}
```

Чунончӣ аз дида мешавад, дар функсияи Cube () функсияи Kvadrat истифода шудааст. Дар функсияи DarPanj() бошад, ҳам функсияи Cube () ва ҳам Kvadrat() истифода шудааст. Натиҷаи ин барнома ба сурати зерин аст:

Қаламрави тағйирёбандаҳо

Дар забони C++ ду намуди тағйирёбандаҳо ба кор бурда мешаванд:

1. Тағйирёбандаҳои маҳаллӣ(local variables ё private variables)
2. Тағйирёбандаҳои умумӣ(global variables)

1) Тағйирёбандаҳое, ки дар дохили як функсия ё як блок муаррифӣ карда мешаванд, тағйирёбандаҳои маҳаллӣ номида мешаванд. Ин гуна тағйирёбандаҳо фақат дар дохили як функсия шинохта мешаванд. Онҳоро ҳатто функсияи асли ҳама намешиносад, аз ин хотир онҳоро тағйирёбандаҳои маҳаллӣ ё хусусӣ меноманд. Дар забони C++ ин навъи тағйирёбандаҳоро тағйирёбандаҳои худкура (automatic) низ мегуянд.

Тағйирёбандаҳои мазкур дар вақти иҷроиши барнома, ҳангоме ки идоракуни дохили он блок ё функсияе, ки дар он муаррифӣ шудаанд, мегардад, арзи вучуд мекунад ҳалос. Ҳарчанд ки дар блокҳо ё функсияҳои дигар бо ин ном тағйирёбандаҳо мавҷуд бошанд, онҳо байни ҳам ҳеҷ робитае надоранд. Ин ба монанди он аст, ки дар як шаҳр ду одам бо як ном мавҷуданд ва бо ҳамдигар ҳеҷ робитае надоранд. Мисол:

```
Func1()
{
    int d=12;
}
Func2()
{
    int d=-25;
}
```

Тавре ки аз ин мисол айён аст, ду функсия тағйирёбандаҳои бо номи d муаррифӣ ва истифода шудаанду миқдорашон аз ҳамдигар фарқ мекунад. Ин тағйирёбандаҳо бо як ном муаррифӣ шудаанд, вале байни ҳам ҳеҷ робита ё алоқамандие надоранд, яъне тағйирёбандаи d-и функсияи дуюм аз тағйирёбандаи d-и функсияи якум фарқ мекунад. Онҳо метавонанд аз навъҳои гуногун иборат бошанд.

Тағйирёбандаҳои маҳаллӣ метавонанд дар як функсия дар дохили ҳар блокҳо тавсиф карда шаванд. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    double x, S;
    cout << "x = "; cin >> x;
    S = x / 2;
    cout << "S = " << S << endl;
    {
        int x, S;
        cout << "local variables " << endl;
        cout << "x = "; cin >> x;
        S = 2 * x;
        cout << "S = " << S << endl;
    }
}
```

```
cout << "\nS = " << S << endl;
return 0;
}
```

Дар ин чо чи хеле ки гуфтем, дар дохили блок тағйирёбандаи x ва S аз навъи бутун ҳамчун тағйирёбандаи маҳаллӣ муаррифӣ шудаанд. Вақте ки иҷроиши барнома дохили блок мешавад, онҳо вучуд доранд, вале дар ҳолати берун шудан аз блок дигар онҳо вучуд надоранд ва ҳофиза аз онҳо ҳолӣ карда мешавад. Аз натиҷаи барнома, ки сурати зеринро дорад метавон сарътар мушоҳида кард:

```
x=12
S=6
local variables
x=8
S=16
S=6
```

Тағйирёбандаҳои умумӣ бошанд, тағйирёбандаҳои ҳастанд, ки дар тамоми тӯли барнома шинохта шуда, ҳар кучо метавонанд мав-риди истифода қарор гиранд. Инчунин миқдори худро дар тамоми вақти иҷроиши барнома нигоҳ доранд. Тағйирёбандаҳои мазкур пеш аз функсияи аслии муаррифӣ карда мешаванд. Барои ба онҳо зудтар дастрас шудан ва аз тағйирёбандаҳои маҳаллӣ фарқ карданашон, аз оператори ягонаи иҷозати ҳудуди фаъолият :: истифода бурда мешавад.

Мисол. Барномае нависед, ки дар он аз тағйирёбандаҳои маҳаллӣ истифода шуда бошад.

```
#include <iostream>
using namespace std;
const double PI = 3.14159;
int main(){
    float PI = 3.14;
    cout << "Qimati Localii PI = " << PI << endl;
    cout << "Qimati globalii PI = " << ::PI << endl;
    return 0;
}
```

Тавре ки аз ин мисол маълум гардид, ду тағйирёбанда бо номи PI муаррифӣ шудаанд, ки якеаш тағйирёбандаи маҳаллӣ аз навъи float буда, дорои қимати PI=3.14 мебошад. Тағйирёбандаи дигар бо ин ном ҳамчун тағйирёбандаи умумӣ аз навъи double муаррифӣ шуда, дорои қимати PI=3.14159 мебошад. Дар операторҳои хоричкунӣ нахуст қимати тағйирёбандаи PI, ки аз навъи маҳаллӣ аст, дар саҳифаи тасвир хорич карда шудааст. Дар оператори хоричкунии дуввум бошад, қимати тағйирёбандаи умумии PI бо ёрии оператори ягонаи иҷозати ҳудуди фаъолият (::) дар саҳифаи тасвир навишта шудааст.

Тағйирёбандаҳои умумӣ корбурди зиёде доранд, вале бар замми ин аз норасоӣ ҳолӣ нестанд. Бинобар ин, бо далелҳои зерин бояд аз истифодаи ин навъ тағйирёбандаҳо худдорӣ кард.

1. Онҳо дар тамоми муддати иҷрои барнома ҳофизаро банд меку-нанд.

2. Истифода аз тағйирёбандаҳои сатҳи паҳншуда мумкин аст ба баъзе сабабҳо шинохта нашаванд, дар натиҷа барнома ба хатогӣ рӯбарӯ мешавад.

3. Дар барномаҳои бузург як тағйирёбанда лозим мешавад, ки қиматашро дар баъзе қисмат дигар карда, дар ҷои дигар боз ба қимати аввала соҳиб шавад. Тағйирёбандаҳои умумӣ бошанд, Тавре ки гуфтем, дар тамоми тӯли барнома қимати худро тағйир намедиханд.

Class-ҳои ҳофиза

Class-ҳои ҳофиза қаламрав ва бақои як тағйирёбандаро дар як барнома муайян мекунанд, яъне дар кучо дастрас будан ва дар кучо дастрас набудани тағйирёбандаро муайян мекунанд. Дар забони C++ чунин class-ҳо и ҳофиза вучуд доранд:

№	Номи class	Истифодаи class дар забони C++
1	Синфи ҳофизаи худкор (automatic)	auto
2	Синфи ҳофизаи доимӣ ва ё собит (static)	static
3	Синфи ҳофизаи registerӣ (register)	register
4	Синфи ҳофизаи хориҷӣ (external)	external

Тарзи истифодаи ин class-ҳо ба сурати зерин аст:

```
auto int variable1, variable2,  
static int variable =10;  
register int variable;
```

Дар сатри дуюм илова бар он ки variable аз навъи бутун муаррифӣ шудааст, ба он қимати аввала бахшида шудааст. Ҳоло барои пурратар шинос шудан ҳар яке аз ин class-ҳоро дар алоҳидагӣ дида мебароем.

Синфи ҳофизаи худкор (automatic)

Тағйирёбандаҳои синфи худкор (automatic) ҳамеша дар дохили як функсия муаррифӣ ва истифода мешаванд, яъне қаламрави исти-фодаи онҳо танҳо дохили як функсия ва ё як блок мебошанд. Бинобар ин, бо ин ном тағйирёбандаҳое, ки дар дигар ҷойҳо муаррифӣ мешаванд, аз ҳамдигар ҷудо буда мустақил мебошанд.

Ба ҳар ҳол, Тавре ки гуфтем, тағйирёбандаҳои худкор ба берун шудан аз функсия ё блоке, ки дар дохили он муаррифӣ шудаанд, миқдори худро аз даст медиҳанд. Аз мантиқи барнома дида мешавад, ки ҳар гоҳ ки функсия ё блоке вобаста ба он иҷро гардад, тағйир-ёбандаи худкорӣ боз аз сари нав шинохта мешавад.

Мисол. Барномае нависед, ки дар он аз тағйирёбандаҳои навъи худ-кора истифода карда шавад.

```
#include <iostream>  
using namespace std;  
int main(){  
    int a = 25;  
    cout << a << endl;
```

```

{
    auto int a = -45;
    cout << "auto a = " << a << endl;
}
cout << a << endl;
return 0;
}

```

Тавре ки айён аст, дар ин мисол тағйирёбандаи *a* аввал аз навъи бутун муаррифӣ гардидааст ва дар дохили як блок низ бо ин ном тағйирёбандаи худкоре муаррифӣшудааст. Вақте ки идоракунии барнома вориди блок мегардад, қимати ин тағйирёбанда шинохта мешавад ва ҳангоми хориҷ шудан аз блок қимати ин тағйирёбандаи дигар мешавад ва боз қимати аввалии *a* дар саҳифаи тасвир хориҷ карда мешавад.

Синфи ҳофизаи доимӣ (static)

Тағйирёбандҳои доимӣ тағйирёбандҳои мебошанд, ки дар тамоми барнома то замоне ки барнома истифода мешаванд, қимати худро тағйир намедиханд. Тағйирёбандҳои синфи мазкур, Тавре ки тағйирёбандҳои синфи худкор буданд, дар дохили як функсия муаррифӣ ва истифода мешаванд, ба ин фарқият, ки пеш аз онҳо каллимаи калидии *static* навишта шуда, барои онҳо низ қимати аввала бахшида мешаванд. Дар ҳолати ба онҳо набахшидани қимати аввала ба таври автоматӣ барои худ қимати нол қабул мекунанд. Онҳо метавонанд ба тағйирёбандҳои беруни ҳамном бошанд.

Мисол. Барномае нависед, ки дар он тағйирёбандаҳои *class static* истифода шаванд.

```

#include <iostream>
using namespace std;
float a = 10, b = 15;
int main(){
    cout << "a = " << a << endl;
    cout << "b = " << b << endl;
    static int a = 123, b;
    cout << "static a = " << a << endl;
    cout << "static b = " << b << endl;
    return 0;
}

```

Тавре ки аз ин мисол маълум шуд, тағйирёбандҳои *a* ва *b* аз навъи умумӣ муаррифӣ шуда, дар функсияи асосӣ хориҷ карда шудаанд, сипас, дар дохили ҳамон функсия ду тағйирёбанда ба ҳамин ном аз синфи *static* муаррифӣ шуда, ба якеаш қимати аввала бахшида шуда, ба дигараш қимате бахшида нашудааст. Бинобар ин, ба таври автомати қимати 0 қабул мекунанд. Натиҷаи барнома намуди зеринро дорад:

```

a=10
b=15
static a=123
static b=0

```

Синфи ҳофизаи registerī (register)

Синфи ҳофизаи registerī (register)-як навъи дигари мушах-хаскунии ҳофиза мебошад, ки он фақат барои тағйирёбандаҳои навъи бутун (int) ва рамзӣ (char) хос буда, бештар дар операторҳои даврӣ барои зуд иҷро намудани давр истифода мешаванд. Тағйирёбандаҳои навъи мазкур имконият медиҳанд, ки ба ҷойи ҳофизаи компютер, ки дар он ҳамаи тағйирёбандаҳо нигоҳ дошта мешаванд, ҳофизаи registerī (яъне ҳофизаи худашон) (CPU) истифода шавад. Дар ин сурат амалиёт болои ин навъ тағйирёбандаҳо хеле дуруст ва саҳеҳ амалӣ мегардад. Аз сабаби он ки CPU маҳдуд аст, миқдори барномаҳои, ки дар онҳо тағйирёбандаҳои синфи мазкур истифода мешаванд, кам аст. Инчунин ин навъ синфи ҳофиза танҳо барои тағйирёбандаҳои маҳалли хос мебошад.

Мисол. Барномае нависед, ки дар он тағйирёбандаҳои классии register истифода шаванд.

```
#include <iostream>
using namespace std;
int main(){
    register int k;
    int S = 0;
    for (k = 0; k <= 100; k++)
        S += k;
    cout << "S = " << S;
    return 0;
}
```

Натиҷаи ин барнома S=5050 мешавад.

Синфи ҳофизаи хориҷӣ (external)

Тавре ки қаблан гуфтем, барои суръати иҷроиши барномаҳо зиёд намудан дар забони C++ барномаи бузургро ба якчанд қисмҳо бо номи unit-воҳид ё modul чудо намуда, дар дар файлҳои чудогона қарор дода, баъдан якҷоя намуда тарҷума (Compile) мекунанд. Дар ин равиш агар бихоҳем, ки тағйирёбандаҳоро дар воҳиди аслии муаррифӣ кунем ва онҳоро дар функсияҳои фаръӣ истифода барем (бе он ки дар функсияҳои фаръӣ барои онҳо ҳофиза ихтисос диҳем), зарурат мешавад, ки онҳоро аз навъи хориҷӣ муаррифӣ кунем. Барои дар функсияи лозимӣ онҳоро истифода бурдан, пеш аз навъи тағйирёбандаҳо, калимаи калидии extern навишта мешавад ва он ба компилятор мегӯяд, ки тағйирёбанда аз навъи берунӣ аст ва барои он ҳофиза чудо карда нашавад. Агар як тағйирёбандаи хориҷӣ дар дохили функсияе, ки тағйирёбандаҳои умумӣ дар онҳо муаррифӣ шудаанд истифода шавад, пеш аз он навиштани калимаи extern зарур намебошад. Вақте ки компилятор ба мавқеъе, ки дар он тағйирёбанда муаррифӣ нашудааст, расад ташхис мекунад, ки оё он аз навъи умумӣ аст, агар бошад, онро ҳамчун extern дар назар мегирад.

Агар тағйирёбандаҳои хориҷӣ ба намуди умумӣ (global) муар-рифӣшуда бошанд, қаламрави истифодаи онҳо аз мавқеи муар-рифӣшуда то охири барнома мебошад. Дар ҳар функсияе, ки мо ба ин навъ тағйирёбандаҳо қимат бахшем, бо баромадан аз функсия низ қимати худро аз даст намедиҳанд. Барои тағйирёбандаҳо навъи мазкур бахшидани қимати аввала хос нест, бо истисноии

навъи умумӣ будани онҳо. Ба ҳар ҳол, муаррифӣ намудани функсия агар пеш аз муаррифӣ намудани тағйирёбандаҳои хориҷи биёяд, бояд тағйирёбандаҳои мазкур дар дохили ин навъи функсия ҳамчун хориҷӣ муаррифӣ гардад.

Мисол. Дар ин мисол тарзи истифодаи тағйирёбандаҳои классии `extern` истифода шудаанд.

```
#include <iostream>
using namespace std;
extern int d=10;
int main(){
    cout<<"d="<<d<<endl;
    {
        d=20;
        cout<<"d="<<d<<endl;
    }
    cout<<"d="<<d<<endl;
    {
        d=25;
        cout<<"d="<<d<<endl;
    }
    cout<<"d="<<d<<endl;
    return 0;
}
```

Тавре ки дида мешавад, дар ин мисол тағйирёбандаи `d` аз навъи умумӣ ва хориҷӣ муаррифӣ гардида, дар дохили функсия бо ёрии оператори ҳоичкунӣ ба чоп дода шудааст, сипас, дар дохили блоки яқум ба он қимати `d=20` бахшида шуда, бо ёрии оператори хориҷкунӣ низ ба чоп дода шудааст. Берун аз блок бори дигар бо ёрии оператори хориҷкунӣ `d`-ро ба чоп дода шудааст ва Тавре ки дида мешавад, қимате, ки дар дохили блок омадааст, ба қимати пас аз он омада яқхела мебошад. Баъд аз ин, бори дигар дар дохили блоки дигаре ба тағйирёбандаи мазкур қимати `d=25` бахшида, бо ёрии оператори хориҷкунӣ ба чоп дода шудааст. Пас аз ин функсия бошад, қимати `d` низ бо ёрии оператори хориҷкунӣ ба чоп дода шудааст ва Тавре ки маълум мешавад, дар ин ҷо қимати `d` ба қимати аз ҷама охирон бахшидаи он баробар мебошад. Натиҷаи барнома шакли зеринро дорад:

```
d=10
d=20
d=20
d=25
d=25
```

Ба аргументҳои функсия бахшидани қимати аввала

Ҳангоми даъвати функсия аргументҳое, ки ба он равона карда мешаванд, бояд лорой ягон қимат бошанд, вагарна барнома ба хатогӣ рӯбарӯ мешавад. Дар забони C++ барномасоз метавонад ҳангоми пешнамунасосӣ ба аргументҳои яунксия қимати ибтидои бахшад. Пас, иҷрои ин амал агар дар вақти даъвати функсия аргументҳо нишон дода нашуда бошанд ҳам, барнома ба хатогӣ рӯбарӯ

намешавад. Функция ба таври автоматӣ ҳамон қимати аргументҳоеро, ки ба он пештар бахшида будем, истифода мекунад.

Мисол. Функцияе нависед, ки ҳаҷми параллелопипедро ҳангоми дода шудани баландӣ, дарозӣ ва васеъгии он ҳисоб кунад.

```
#include <iostream>
using namespace std;
int HajmParall(int darozi=1, int bar=1,int balandy=1);
int main(){
    cout<<"Hajmi Parallelopiped bo qimathoi qably :"<<HajmParall()<<"\n";
    cout<<"Hajmi Parallelopiped bo doda shudani darozi va bar
="<<HajmParall(8,5)<<"\n";
    cout<<"Hajmi Parallelopiped bo doda shudani darozi,bar va
balandi:"<<HajmParall(10,5,2)<<" \n ";
    return 0;
}
int HajmParall(int darozi, int bar,int balandy){
    return darozi* bar* balandy;
}
```

Тавре ки аз ин ҷо дида мешавад, ҳангоми пешнамунасозии функция барои аргументҳои он қимати 1 бахшида шудааст. Дар оператори хориҷкунии яқум ҳарчанд ки функция дорои аргумент намебошад, вале ба таври автоматӣ қиматҳои авваларо қабул мекунад. Дар оператори хориҷкунии дуввум бошад, танҳо дарозӣ ва бари параллелопипед нишон дода шудаасту ҳалос, баландии он ба таври автоматӣ қимати авваларо қабул мекунад. Натиҷаи барнома шакли зеринро дорад:

```
Hajmi Parallelopiped bo qimathoi qably =1
Hajmi Parallelopiped bo doda shudani darozi va bar =40
Hajmi Parallelopiped bo doda shudani darozi,bar va balandi =100
```

Сарборгузории функцияҳо

Забони C++ имконият медиҳад, ки якчанд функцияҳоро бо як ном ва набори аргументҳои гуногунавъ муаррифӣ кунем. Чунин амалро сарборигузории функция меноманд. Компилятор ҳангоми даъвати функция ба навъи аргументҳои он назар меафканад. Агар аргументҳо ба кадом варианти функция, ки мувофиқ оянд, ҳамон варианти функция даъват карда мешавад. Мисол. Функцияе нависед, ки квадрати адади навъи дилхоҳро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int Kvadrat(int n){
    return n*n;
}
double Kvadrat(double n){
    return n*n;
}
float Kvadrat(float n){
    return n*n;
}
```

```
int main(){
    cout<<"Kvadrati int="<<Kvadrat(3)<<endl;
    cout<<"Kvadrati float="<<Kvadrat(0.5)<<endl;
    cout<<"Kvadrati double="<<Kvadrat(3.512)<<endl;
    return 0;
}
```

Тавре ки дида мешавад, дар ин ҷо се функсия бо як ном вале бо аргументҳои гуногунавъ муаррифӣ шудаанд. Дар функсияи асосӣ бо ёрии оператори хоричкунӣ ин функсия бо аргументҳои гуногун ба ҷоп дода шудааст. Натиҷаи барнома шакли зеринро дорад:

```
Kvadrati int=9
Kvadrati
float=0.25
Kvadrati
double=12.3341
```

Қолаби функсияҳо

Тавре ки аз мавзӯи «сарборгузории функсияҳо» мушоҳида кардем, барои онуи функсиямон болои навъи дилхоҳ кор кунад, зарур аст барои ҳар як навъ варианти ба он вобастаи функсия навишта шавад. Ин амал хело хуб аст, вале вақти зиёд ва ҷои зиёдро банда намуда, матни барномаро зиёд мекунад. Барои ин амалро ба таври сода иҷро намудан, ки камтар бошад, аз қолаби функсияҳо, ки дар версияҳои охири C++ ба вуҷуд омаданд, истифода бурдан мумкин аст. Қолабҳои функсия матни барномаро хело кӯтоҳ намуда, аз вуқуи хатогиҳо пешгири мекунанд.

Мисол. Функсияе нависед, ки аз се адади додашуда калонтаринашро барорад. Ададҳои додашуда метавонанд аз дилхоҳ навъи стандартӣ бошанд.

Ҳал. Аввал масъалаи додашударо тавассути сарборигузории функсияҳо ва баъд бо истифода аз қолабҳо менависем, то ин ки ба хонанда амалқарди ин ду механизм ошкор ва фаҳмо гардад. Ба намуди сарборигузори лозим аст, ки барои ҳар навъи стандарти варианти ба он мувофиқи функсияро нависем.

```
#include <iostream>
using namespace std;
int Max(int a, int b,int c){
    int max=a;
    if (max<b)
        max=b;
    if (max<c)
        max=c;
    return max;
}
double Max(double a, double b,double c){
    double max=a;
    if (max<b)
        max=b;
    if (max<c)
        max=c;
}
```



```

        return max;
    }
    int main()
    {
        cout<<"Max int="<<Max(4,5,8)<<endl;
        cout<<"Max double="<<Max(5.8,9.75,0.8)<<endl;
        return 0;
    }

```

Тавре ки дида мешавад, дар ин ҷо барои ду навъи додаҳо (int ва double) варианти функсияро навистем, агар мо барои ҳамаи навъи додаҳо функсия нависем, хело бузург мешавад ва идоракунии он мушкил мегардад. Аз ин рӯ зарурат мешавад, ки роҳеро ҷустуҷӯ кунем, ки матнӣ барномаи мо кӯтоҳ гардад, то барои навиштани он вақти камтар сарф шавад. Роҳи аз ҳама хуб истифодаи қолибҳои функсия мебошад. Ҳамаи қолабҳои функсия бо калимаи калидии «template» оғоз мешаванд. Тарзи муаррифи намудани қолаби функсия бошад, ба сурати зерин аст:

```

template <class Type> //сарлавҳа
{
    Тани функсия
}

```

Дар ин ҷо template калимаи калидӣ, class Type-навъи класс ё навъи аргументҳо ва натиҷаи функсия. Тани функсия бошад, маҷмӯи амалҳои иҷроенамудаи функсия мебошанд. Ба сифати мисол метавонем барномаи каблиро ба сурати зерин навист.

```

#include <iostream>
using namespace std;
template <class T>
T Max(T a, T b, T c){
    T max=a;
    if(max<b)
        max=b;
    if(max<c)
        max=c;
    return max;
}
int main(){
    cout<<"Max int="<<Max(4,5,8)<<endl;
    cout<<"Max double="<<Max(5.8,9.75,0.8)<<endl;
    cout<<"Max char="<<Max('x','y','Z')<<endl;
    return 0;
}

```

Дар ин ҷо template- калимаи калидӣ, <class T>- номи класс, ва T Max(T a, T b, T c)-бошад, функсияе бо номи Max аз навъи қолаб мебошад. Ифодаи T max=a бошад, тағйирёбандаи max-ро аз навъи қолаб муаррифи мекунад. Тавре ки дида мешавад, дар ин ҷо бо як қолаб калонтарин қимати дилоҳ навъи додаҳо муайян карда мешавад. Ин амал Тавре ки аён аст, вақти камтарро гирифта матнӣ барномаро аз зиёдшавишироҳ медорад.

Саволҳо барои мустаҳкамкунӣ

1. Функсия гуфта, чиро дар назар доред ва мақсади истифодаи он дар чист?
2. Чӣ тавр дар барнома функсия тавсиф (муаррифӣ) карда мешавад?
3. Функсия чӣ тавр қиматеро бармегардонад?
4. Фарқи функсияи асли аз фаръӣ дар чист?
5. Оператори return дар функсияҳо барои чӣ хизмат мекунад?
6. Функсияҳои бозгаштӣ гуна функсияанд?
7. Чаро функсияро пешнамуна месозанд?
8. Оё метавонем дар як барнома якчанд функсияро истифода барем?
9. Тағйирёбандаҳо чанд хел мешаванд?
10. Чанд синфи ҳофиза мавҷуд аст?
11. Синфи ҳофизаи register-и аз синфи ҳофизаи auto бо чӣ фарқ мекунад?
12. Амалгари иҷозати ҳудуди фаъолият (::) барои чӣ хизмат мекунад?
13. Сарборӣ намудани функсия чӣ маъно дорад?
14. Сабаб дар чист, ки аз қолабҳо истифода мекунад?
15. Чӣ гуна функсияҳо қимат барнамегардонад?
16. Магар метавонем навъи аргументҳои функсияро тавсиф накунем?
17. Тағйирёбандаи маҷозӣ чист?
18. Оё хангоми пешнамунасозии функсия ба аргументҳои он метавонем қимати аввала махшем?
19. Қимати аввала бахшидан ба аргументҳои функсия оё кори зарур аст?
20. Тағйирёбандаҳое, ки дар дохили як блок ва ё функсия тавсиф мегарданд, чӣ гуна тағйирёбандаанд?

Намунаи ҳалли масъалаҳо

1. Функсияе нависед, ки суммаи ададҳои токи аз 1 то n-ро ҳисоб кунад.

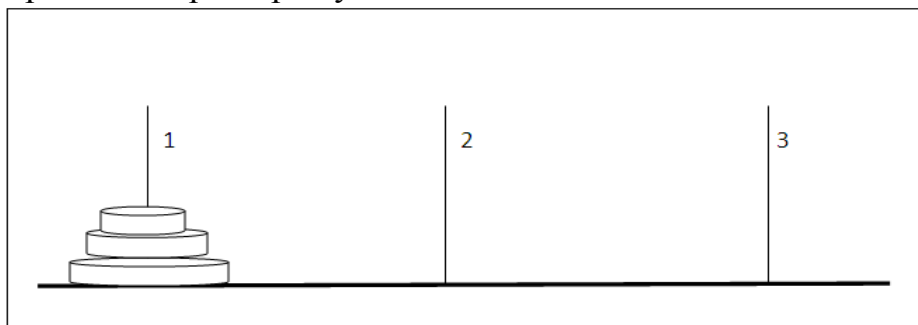
```
#include <iostream>
using namespace std;
long int summa(int n){
    long int S=0; int i ;
    if (n>0)
        for ( i=1;i<=n;i++){
            if (i%2!=0)
                S+=i ;
        }
    return S;
}

int main(){
    int n;
    cout<<"n=";<<cin>>n;
    cout<<"Summa adadhoi toqi az 1 to "<<n<<" = "<<summa(n)<<endl;
    return 0;
}
```

2. Функсияе нависед, ки расми секунҷаи росткунҷаро бо ёрии рамзи ситорача «*» тасвир кунад.

```
#include <iostream>
using namespace std;
void Sekunja(){
    cout<<" *";
}
int main(){
    int i,j;
    for(i=0;i<=5;i++)
    {
        for (j=0;j<=i;j++)
            Sekunja();
        cout<<endl;
    }
    return 0;
}
```

3. Дар асри XIX дар Европо бозие бо номи «Бурҷҳои Ханой» пайдо шуд. Мақсади бозӣ аз он иборат буд, ки се то сутунча дода мешаванд. Дар сутунҷаи якум Ҳалқаҳо бо тартиби калоншавиашон чида мешаванд. Талаб карда мешавад, ки Ҳалқаҳоро ба сутунҷаи дигар (сеюм) тавре гузаронед, ки тартиби ҷойгиршавиашон дигар нашавад, яъне ҳеҷ гоҳ Ҳалқаи калон дар болои Ҳалқаи хурд қарор нагирад. Илова бар ин, бояд Ҳалқаҳо якто-якто кўчонида шаванд. Дар поён барои се Ҳалқа расми бозӣ оварда шудааст. Функсияи бозгаштиё нависед, ки кори ин бозиро иҷро кунад.



Матни барнома:

```
#include <iostream>
using namespace std;
int n;
void BurjHano(int m, int a,int b, int c){
    if (m>0){
        BurjHano(m-1, a , c, b);
        cout<<"Halkai raqami " <<m<<" az sutuni " << a<<" ba sutuni
"<<b<<" kuchonida shud: " <<endl;
        BurjHano(m-1, c , b, a);
    }
}
int main(){
    cout<<"n=";<<cin>>n;
    BurjHano(n, 1 , 3, 2);
}
```

```

return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

n=3
Halkai raqami 1 az sutuni 1 ba sutuni 3 kuchonida shud:
Halkai raqami 2 az sutuni 1 ba sutuni 2 kuchonida shud:
Halkai raqami 1 az sutuni 3 ba sutuni 2 kuchonida shud:
Halkai raqami 3 az sutuni 1 ba sutuni 3 kuchonida shud:
Halkai raqami 1 az sutuni 2 ba sutuni 1 kuchonida shud:
Halkai raqami 2 az sutuni 2 ba sutuni 3 kuchonida shud:
Halkai raqami 1 az sutuni 1 ba sutuni 3 kuchonida shud:

```

4. Функцияе нависед, ки насаб ё номи донишҷӯёнро аз рӯи алифбо мураттаб намояд.

```

#include <iostream>
#include <stdlib.h>
#include <string.h>
using namespace std;
void buble_sort(char *array[], int n){
    char *temp;
    int i, j;
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            if (strcmp(array[i], array[j])<0)
            {
                temp=array[i];
                array[i]=array[j];
                array[j]=temp;
            }
}
int main(){
    char *mass[]= {"Zaripov" , "Alimov", "Fozilov",
"Qamarov","Nuriddinov", "Davlatov", "Saidov"};
    int i;
    buble_sort(mass, 7);
    for(i=0;i<7;i++)
        cout<<mass[i]<<" " ;
    return 0;
}

```

5. N адад пайҳам аз клавиатурӣ дохил карда мешаванд. Функция нависед, ки қимати миёнаи арифметикии ададҳои содаи воридшуда ва функцияи дигар нависед, ки қимати миёнаи ададҳои содаи воридшударо ҳисоб кунад. Агар чунин ададҳо набошанд, пайғоми муносибе чоп кунад.

```

#include <iostream>
#include <cmath>
using namespace std;
unsigned int sodda(unsigned int N){
    int i,k;
    for(k=1,i=2;i<=N/2;i++)

```

```

        if (N%i==0){ k=0;break;
    }
    return k;
}
unsigned int Mukammal(unsigned int N){
    unsigned int i,S;
    for (S=0,i=1;i<=N/2;i++)
        if (N%i==0) S+=i;
    if(S==N)return 1;
    else
        return 0;
}
int main(){
    unsigned int i,N,X,S,ks,km;
    long int P;
    cout<<"N=";cin>>N;
    for(ks=km=S=0,P=1,i=1;i<=N;i++)
    {
        cout<<"X=";cin>>X;
        if (sodda(X))
        {
            ks++;    //Ҳисобкунаки миқдори ададҳои сода
            P*=X;    //Ҳосили зарби ададҳои сода
        }
        if (Mukammal(X))
        {
            km++;    //Ҳисобкунаки ададҳои мукамал
            S+=X;    //Суммаи ададҳои мукамал
        }
    }
    if (ks>0)
        cout<<"MGeom="<<pow(P,(float)1/ks)<<" \n ";
    else
        cout<<"Adadi sodda nest!\n";
    if (km>0)
        cout<<"MArif="<<(float)S/km<<" \n ";
    else
        cout<<"Adadi mukammal nest!\n";
    cout<<sodda(100);
    return 0;
}

```

6. Пайдарпайии ададҳо, ки 0 охири пайдарпайӣ аст, ворид карда мешаванд. Барномае нависед, ки суммаи ададҳои сода ва суммаи ададҳои ғайри содаи дохили пайдрпайиро ҳисоб кунед. Барои муайянкунии ададҳои сода функсия нависед.

```

#include <iostream>
#include <cmath>
using namespace std;
unsigned int Sodda(unsigned int N){

```

```

int i,k;
for(k=1,i=2;i<=N/2;i++)
    if(N%i==0)
        k=0;
return k;
}
int main(){
    unsigned int X, M=0, S=0;
    for(cout<<"X=",cin>>X;X!=0;cout<<"X=",cin>>X)
    {
        if (Sodda(X))
            S+=X;
        else
            M+=X;
    }
    cout<<"Murakkab="<<M<<"\nSodda="<<S<<endl;
    return 0;
}

```

7. Функсияе нависед, ки бо дода шудани радиуси давра (R) дарозии давра (L)-ро ҳисоб кунад.

```

#include <iostream>
using namespace std;
double DarDavra(double R){
    return 3.14*2*R;
}
int main(){
    double R;
    cout<<"R=",cin>>R;
    cout<<"Darozii davra="<<DarDavra(R)<<endl;
    return 0;
}

```

8. Функсияе нависед, ки бо дода шудани адади бутуни N (N>0), элементҳои пайдарпайии ададҳои ҳақиқии A_k -ро, ки намуди зеринро доранд, барорад.

$$A_0 = 2; A_k = 2 + \frac{1}{A_{k-1}}, \quad k = 1, 2, \dots$$

```

#include <iostream>
using namespace std;
double PP(int N){
    if (N==0)
        return 2;
    else
        return 2+1/(PP(N-1));
}
int main(){
    int n;
    cout<<"n=";cin>>n;
    for (int i=0;i<=n;i++)

```

```

        cout<<i << "." << " Paydarpai=" << PP(i) << endl;
    return 0;
}

```

9. Функцияе нависед, ки КТУ-и ду ададро аз рӯи алгоритми Евклид ҳисоб кунад.

```

#include <iostream>
#include <cmath>
using namespace std;
int KTU(int a,int b){
    if(b==0)
        return a;
    else
        return KTU(b,a%b);
}
int main(){
    int a,b;
    cout<<"a=";<<cin>>a;
    cout<<"b=";<<cin>>b;
    cout<<"KTU("<<a<<","<<b<<")="<<KTU(a,b);
    return 0;
}

```

10. Бо ёрии функцияе қимати синуси гипорбиликии x -ро ҳисоб кунед. Функцияи гиперболи хангоми ба қатор паҳн кардан намуди зеринро дорад:

$$\text{sh}(x) = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots = \sum_{n=0}^{\infty} \frac{x^{2n+1}}{(2n+1)!}$$

```

#include <iostream>
#include <cmath>
using namespace std;
const int N=100;
double sh(double z){
    int i;
    double S=z,q=z;
    for (i=1;i<=N;i++){
        q*=z*z/(2*i)/(2*i+1);
        S+=q;
    }
    return S;
}
int main(){
    double x;
    int i;
    cout<<"x=";<<cin>>x;
    cout<<"sh("<<x<<")="<<sh(x)<<endl;
    cout<<"sinh("<<x<<")="<<sinh(x)<<endl;
    return 0;
}

```

11. Ҳисоб кунед.

$$\prod_{i=1}^{\infty} (1 + x^{2^i}) = \frac{1}{1-x} \quad (|x| < 1)$$

```
#include <iostream>
#include <cmath>
using namespace std;
double Zarb(double x, int i){
    if (i>0)
        return Zarb(x,i-1)*(1+pow(x,pow(2,i)));
    else
        return 1+x;
}
int main(){
    double x;
    int n,i;
    cout<<"x=";cin>>x;
    cout<<"i=";cin>>i;
    cout<<"Qimati Zarb dar tobe="<<Zarb(x,i)<<endl;
    cout<<"Qimati Zarb be tobe="<<1/(1-x)<<endl;
    return 0;
}
```

12. Функцияе нависед, ки суммаи ададҳои содаи порчаи [1...n]-ро ҳисоб кунад.

```
#include <iostream>
#include <cmath>
using namespace std;
long SummaSodda(int n){
    int i,j,S,k;
    for (S=0,i=2;i<=n;i++){
        for (k=0,j=2;j<=(int) sqrt(i);j++)
            if (i%j==0)
                k++;
        if(!k)
            S+=i;
    }
    return S;
}
int main(){
    int n,i;
    cout<<"n=";cin>>n;
    cout<<SummaSodda(n)<<"\n ";
    return 0;
}
```

13. Шаблони функцияе нависед, ки КТУ-и ду қимати додасударо ҳисоб кунад.

```
#include <iostream>
using namespace std;
template <class A>
A KTU (A a,A b){
```



```

        if(b==0)
            return a;
        return KTU(b,a%b);
    }
int main(){
    int a,b;
    cout<<"a=";cin>>a;
    cout<<"b=";cin>>b;
    cout<<KTU(a,b)<<" \n ";
    return 0;
}

```

14. Функцияи $f(x)=\sin x+\cos x$ дода шудааст. Барномае нависед, ки бо дода шудани қиматҳои a ва b қимати $f(f(a+b))$ -ро ҳисоб кунад. Барои ҳалли ин мисол функция нависед.

```

#include <iostream>
#include <cmath>
using namespace std;
double F(double x){
    return pow(x,(double)1/3)+pow(x,(double)4);
}
int main(){
    double a,b,P;
    cout<<"a=";cin>>a;
    cout<<"b=";cin>>b;
    P=F(a+b);
    cout<<F(P)<<" \n ";
    return 0;
}

```

15. Ду адад ба ҳам «дўст» «номида мешаванд, агар яке аз онҳо ба суммаи тақсимкунандаҳои дигараш (бе худи адад) баробар бошад. Функцияе нависед, ки ададҳои «дўст»-и порчаи $[2...n]$ -ро барорад.

```

#include <iostream>
using namespace std;
int main() {
    int n,i,S,k,j;
    cout<<"n=";cin>>n;
    for (i=2;i<=n;i++)
    {
        S=1;
        for (j=2;j<=i/2;j++)
            if((i%j)==0) S+=i;
        cout<<S<<endl;
        for (k=2;k<=n;k++)
            if ((k==S) && (k!=i) )
                cout<<k<<" " <<i<<endl;
    }
    return 0;
}

```

Масъалаҳо барои кори мустақилона

1. Ҳисоб кунед.

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)!}$$

2. Адади бутуни n ($n > 0$) дода шудааст. Функцияе нависед, ки пайдарпайии ададҳои ҳақиқии A_k -ро чоп кунад.

$$A_0 = 1, A_k = \frac{(A_{k-1} + 1)}{k}, \quad k = 1, 2, \dots$$

3. Адади бутуни n ($n > 0$) дода шудааст. Пайдарпайии ададҳои ҳақиқии A_k намуди зеринро доранд:

$$A_1 = 1, A_2 = 2, A_3 = 3, A_k = A_{k-1} + A_{k-2} - 2A_{k-3}, \quad k = 4, 5, \dots$$

Функцияе нависед, ки пайдарпайии элементҳои A_1, A_2, \dots, A_n -ро чоп кунад.

4. Якчанд адади ҳақиқӣ пасиҳам дохил карда мешаванд. Адади 0 охирин ин пайдарпайи мебошад. Функцияе нависед, ки суммаи қисмҳои касри ин ададҳоро ҳисоб кунад.

5. Координатаҳои қуллаҳои секунҷа $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$ маълуманд. Функцияе нависед, ки масоҳати секунҷаро ҳисоб кунад.

6. Дарозии се порча a, b, c дода шудаанд. Функцияе нависед, ки муайян кунад, ки оё бо ёрии ин порчаҳо секунҷа сохтан мумкин аст.

7. Ҳисоб кунед:

$$P = \begin{cases} (n! + 1)^3, & \text{агар } n \geq 5 \text{ бошад.} \\ \sin(n!), & \text{агар } n < 5 \text{ бошад.} \end{cases}$$

8. Шаблони наивисед, ки бо дода шудани x, y , a ва b ҳисоб кунед $\min(\max(m, n), \max(x, y))$

9. Барои қиматҳои ҳақиқии x ҳисоб кунед

$$\cos x \operatorname{tg}(x+1) + \cos^2(2 + \cos(x+1))$$

10. Ададҳои x, y ва z дода шудаанд. Ҳисоб кунед

$$S = \frac{\max(x, y, z) - 2\min(x, y, z)}{\cos x + \frac{\min(x, y, z)}{\max(x, y, z)}}$$

11. Пайдарпайии ададҳои Фибоначи, ки намуди зеринро доранд, дода шудаанд:

$$\begin{cases} U_0 = 1 \\ U_1 = 2 \\ U_2 = 3 \\ U_n = U_{n-1} + U_{n-2} + U_{n-3} \end{cases}$$

бо додани адади бутуни n ($n \geq 0$) элементҳои n -уми ин пайдарпайиро ёбед.

12. Ҳисоб кунед:

$$1) \sqrt{3 + \sqrt{3 + \sqrt{3 + \dots + \sqrt{3}}}} \quad 2) \sqrt{1 + 2\sqrt{1 + 3\sqrt{1 + 4\sqrt{\dots}}}}$$

13. Ҳисоб кунед:

$$S = \sqrt{1 + (n + 1) \sqrt{1 + (n + 2) \sqrt{1 + (n + 3) \sqrt{1 + \dots}}}}$$

14. . Ҳисоб кунед: $y = a + (a + (a + (a + (a + (a + (a + (a + x)^3)^3)^3)^3)^3)^3)^3$

15. Функцияе нависед, ки ҳангоми дода шудани адади бутуни $1 \leq n \leq 10$ номи рақамро ба чоп диҳад. Масалан:

Доҳилкунӣ 1, 2, 3, ...

Хориҷкунӣ як, ду, се, ...

16. Адади чоррақама дода шудааст. Муайян кунед, ки рақамҳои ин адад квадрати пуррарор ташкил медиҳанд.

17. Се адади бутун дода шудаанд, КТУ-и ин ададхоро ҳисоб кунед.

18. Функцияе нависед, ки дараҷаи адади n -ро ҳисоб кунад. Маълум аст, ки дараҷаи n -и адади x бо ёрии формулаи зерин ҳисоб карда мешавад.

$$x^n = \begin{cases} 1, & \text{агар } n = 0 \text{ бошад} \\ xx^{n-1}, & \text{агар } n \neq 0 \text{ бошад.} \end{cases}$$

Боби 6. Массивҳо (arrays)

Муқаддима

Барои ичрои ягон амал (масалан чамъ) бо ду адади додашуда, се тағйирёбанда гирифта, якеашонро бо a , дигарашро бо b ва натиҷаро бо s ишора мекунем. Ба ҳамин монанд барои се адад аз чор тағйирёбанда истифода бурда мешавад. Акнун бигзор n ($n > 10$) адад дода шуда бошад. Агар барои ҳар яке аз ададҳо як тағйирёбанда гирифта шавад, матни барнома хело калон шуда, барноманавис дилгир мешавад. Аз ин ҷо зарурати истифодаи роҳи дигаре пеш меояд, ки идораи барномаро осон гардонид, матни онро аз зиёдшави нигоҳ дорад. Якчанд роҳи бартараф намудани ин мушкили вучуд дорад, ки аз ҳама роҳи содда истифода аз массивҳо мебошанд.

Ҳар гоҳ ки маҷмӯи додаҳо дорои хусусият ва сифати якхела бошанд, метавонем барои ҳамаи онҳо як ном гузошта, хангоми истифодабарии он номро барои ҳамашон истифода кард. Барои ба ҳар як дода дастрас шудан аз зернавис (subscript) ва ё индекси он истифода мебарем, яъне ҳамаи элементҳои дорои як ном буда, танҳо аз ҳамдигар бо зернависашон фарқ мекунам. Зернавис аз нуқтаи назари риёзӣ рақами тартибии додари ифода мекунад. Масалан, дар дафтари синф ҳар як донишҷӯ дорои ном (фамилия ва ном) ва рақами тартибии ҷойгиршавии номҳо ме-бошад. Аз нуқтаи назари риёзӣ бигзор ба мо пайдарпайии $a_1, a_2, a_3, \dots, a_n$ дода шуда бошад. Дар ин ҷо a -номи массив ва адади дар тарафи рост ва поён навишташуда зернавис ва ё индекс мебошад. Аъзои k -ум гуфта, элемент a_k дар назар гирифта мешавад. Ин гуна маҷмӯъҳо дар забонҳои барноманависи массив (аггау) мегӯянд. Бинобар ин, таърифи илмӣ ва пурраи массивро метавонем ба сурати зерин баён кард:

Таъриф. *Гурӯҳи элементҳои дорои хусусият ва сифати якхелае, ки аз ҳамдигар танҳо бо зернавис фарқ мекунам, массив номида мешавад.*

Аз ин ҷо метавонем маҷмӯи элементҳои якҷинсаро ба як ном муаррифӣ кунем. Ба ғайр аз ин, метавонем чадвали дорои m -сатр ва n -сутунро, ки элементҳои он дорои сифати якхела мебошанд, бо як ном муаррифӣ кунем. Барои дастрасӣ ба элементҳои ин чадвал дар тарафи поёнии ҳар як элемент ду зернавис, ки яке нишондиҳандаи сатр ва дигаре нишондиҳандаи сутун мебошад, истифода мекунем. Масалан, бигзор чадвале дорои n -сатр ва m -сутун буда, каткакҳои он бо додаҳои якҷинса пур карда шуда бошанд. Метавонем барои ҳамаи додаҳои чадвал як ном гузорем. Масалан, a ва барои ба ҳар як элемент дастрас шудан, Тавре ки гуфтем, дар тарафи рост қисми поёнии элементҳо ду зернавис менависем, яъне a_{ij} ($i=1..n$ ва $j=1..m$). Аз нуқтаи назари риёзӣ, массивҳое, ки як зернавис доранд, вектор ва массивҳое, ду зернавис доранд, матритса мегӯянд.

Эълон ва ё муаррифӣ намудани массивҳо (declaration arrays)

Тавре ки қаблан зикр кардем, ҳар гуна дода дар забони C++ пеш аз истифодабарӣ муаррифӣ карда мешавад. Массивҳо низ пеш аз истифодабарӣ муаррифӣ карда мешаванд. Як хусусияте, ки барои дигар додаҳо набуда, танҳо барои массивҳо хос мебошад, ин андозаи массив аст, ки дар вақти

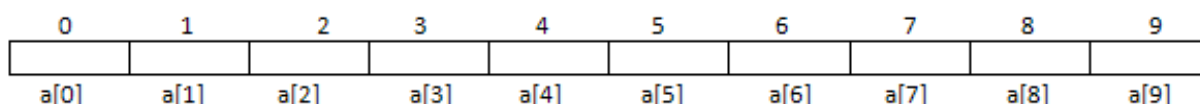
муаррифукунии массивҳо нишон дода мешавад. Андоза ва ё мавқеи ҷойгиршавии массив ҳатман дар дохили қавси фигуравӣ «[]» навишта мешавад. Наҳви эълони массив ба сурати зерин аст:

```
data_type array_name[array_Size]
ё
```

навъи массив номи массив [андозаи массив]

Дар ин ҷо `data_type`-навъи массив (дилхоҳ навъии стандартӣ), `array_name`-номи массив (номи массив бояд монанди дигар тағйирёбандаҳо бо ҳарф сар шавад) ва `array_Size` бошад, андозаи массивро нишон медиҳад. Массиви дорои 10 элементи аз навъи бутун чунин муаррифӣ карда мешавад: `int a[10]`.

Зернависи (Индекси) нахустини массив дар забони C++ аз 0 сар шуда, ҳамаи элементҳои массив ба сурати зерин нишон дода мешаванд:



Ба сифати зернависи массив метавон танҳо аз адади бутун ва ё ифодае, ки натиҷааш адади бутун истифода кард, яъне ифодае, ки аз ададҳои саҳеҳи бутун ва тағйирёбандаҳои бутун, ки бо ёрии операторҳои риёзии + ва - ташкил шудааст, метавонад ба сифати зернависи массив истифода шавад. Масалан, массиви дорои 10 элементро низ метавон ба сурати зерин муаррифӣ кард :

```
const int x=7,y=3;
int a[x+y];
```

Равиши хуби барномасозӣ он аст, ки андозаи массив ҳамчун як адади доимӣ бо ёрии фармони `#define` ва ё `const` муаррифӣ гардад. Навъи массив метавонад дилхоҳ навъҳои стандартӣ бошад ва вобаста ба навъи додасуда барои ҳар як элементи массив дар ҳофиза ҷое банд карда мешавад. Масалан, агар массив аз навъи бутун бошад, барои ҳар як элемент ду байт ҳофиза ҷудо карда мешавад. Аз ин ҷо барои массиве, ки мо қаблан муаррифӣ кардем 20 байт аз ҳофиза банд карда мешавад.

Барои ба элементи дилхоҳи массив дастрас шудан аввал номи массив, баъдан пас аз ном зернависи он дар дохили қавси фигуравӣ навишта мешавад. Масалан, барои элементи чоруми массиви пештар муаррифӣ кардамонро ба тағйирёбандаи `c` бахшидан ифодаи зерин истифода бурда мешавад: `c = a[3]`.

Ин ифодаи як нусхаи элементи чоруми массиви `a`-ро ба тағйирёбандаи `c` мебахшад. Элементҳои массив аз нол рақамгузорӣ мешаванд аз ин рӯ, барои ба элементи `k`-уми массив дастрас шудан ифодаи зерин лозим аст: `c = a[k - 1]`.

Ба массивҳо бахшидани қимати аввала (initialization arrays)

Ҳангоми муаррифукунии массивҳо монанди тағйирёбандаҳои оддӣ метавон ба онҳо қимати аввала бахшид. Элементи аввала дар дохили қавси системави «{ }» навишта шуда, барои аз ҳамдигар ҷудо кардани элементҳо оператори коммо истифода бурда мешавад. Мисол:

```
int array[5]={1,2,3,4,5};
char arraychr[6]={'R','I','y','o','z','I'};
```

Дар ин ҷо - дар сатри якум, массиве аз навъи бутун муаррифӣ карда шудааст, ки дорои 5 элементи бутун мебошад ва дар сатри дуввум бошад, массиве аз навъи рамзи муаррифӣгардидаааст, ки андозааш ба 6 баробар мебошад. Массивҳои навъи рамзиро низ метавон ба таври зерин муаррифӣ кард:

```
char arraychr[6]='RiyozI';
```

Агар андозаи массив аз шумораи элементҳои он зиёд бошад, ба таври автоматӣ элементҳои боқимондаи массив бо нулҳо пур карда мешаванд, инчунин метавон ҳангоми муаррифӣкунӣ ба ҳамаи элементҳои массив қимати нолӣ бахшид. Ба дастури зерин диққат диҳед:

```
int array[10]={0};
```

Агар дар массив ҳангоми муаррифӣкунӣ ҳамаи элементҳои он нишон дода шуда бошанд, ҳолати дар дохили қавси фигуравӣ навиш-тани андозаи он нест, аз ин ҷо мисоли болоро метавонем ба сурати зерин низ муаррифӣкард:

```
int array[]={1,2,3,4,5};
```

Ба ҳамин монанд ба массивҳои дилхоҳ навъ қимати аввала бахшида мешавад.

Доҳилкунӣ ва хоричкунии элементҳои массив

Тавре ки қайд кардем, ҳангоми муаррифӣкунӣ метавон ба массив қимати аввала бахшид, вале баъзан зарурат мешавад, ки элементҳои массивро пас аз ба қор даромадани барнома монанди тағйирёбандаҳои муқаррарӣ дохил намоем. Ҳангоме ки элементҳои массив ба барнома дохил карда мешаванд, зарурате дидани натиҷаи амалиёти иҷрошуда ба он элементҳо пеш меояд, яъне лозим мешавад, ки натиҷаи барномаро бинем. Доҳилкунӣ ва хоричкунии элементҳои массив монанди тағйирёбандаҳои муқаррарӣ сурат мегирад, вале фарқияташ дар он аст, ки дар массивҳо зернавис низ нишон дода мешавад, яъне элементҳои массив аз рӯи шартӣ гузошташуда дохил ва хорич карда мешаванд.

Мисол. Барномае нависед, ки аввал элементҳои массивро дохил намуда, Сипас ба намуди зерин хорич кунад:

```
Array[0]=e1;
Array[1]=e2;
-----
Array[n-1]=eln;
```

```
#include <iostream>
using namespace std;
int main(){
    const int n=5;
    int massiv[n];
    int i;
    for (i=0;i<n;i++)
        cin>>massiv[i];
    for (i=0;i<=n;i++)
```

```

        cout<<"massiv["<<i<<"]="<<massiv[i]<<endl;
    return 0;
}

```

Дар ин ҷо андозаи массив ҳамчун як доимии муаррификардашуда ба 5 баробар мебошад, сипас, то замоне ки шарти оператори даврӣ ҳақ аст, аз клавиатура элементҳои массив дохил карда мешаванд. Ҳангоми ба охир расидани давр идоракунии ба даври дуюм мегузарад ва то он замоне ки давр дуруст аст, элементҳои дохил намуда хориҷ карда мешаванд.

Агар хоҳем, ки элементҳои массив ба намуди сатр хориҷ карда шаванд, он гоҳ дар дастури хориҷкунӣ ба ҷои рамзи идоракунии «\n» рамзи «\t» мегузорем, ба ғайр аз ин, бо ёрии функсияҳои забони C низ метавонем элементҳои массивро хориҷ кунем. Ба барномаи зерин диққат диҳед:

```

#include <iostream>
using namespace std;
int main(){
    const int n=5;
    int massiv[]={1,3,5,7,9};
    for (int i=0;i<n;i++)
        printf("massiv[i]=%i\t",massiv[i]);
    return 0;
}

```

Агар хоҳем, ки элементҳои массив ба намуди сутун хориҷ карда шаванд, ҳангоми хориҷкунӣ ба ҷои рамзи идоракунии «\t» рамзи «\n» мегузорем, дар натиҷа сатре, ки дар он фармони хориҷкунӣ омадааст, намуди зеринро мегирад.

```

printf("massiv[i]=%i\n",massiv[i]);

```

Сумма ва ҳосили зарби элементҳои массив

Бигзор ба мо массиви андозааш муайян дода шуда бошад ва талаб карда шавад, ки суммаи элементҳои онро ҳисоб кунед. Барои иҷрои ин амал дар аввал тағйирёбандае ба номи Summa муаррифӣ карда, ба он ба сифати қимати аввала 0 мебахшем. Баъд аз он, бо ёрии оператори даврӣ ҳар як элементи массивро ба он илова мекунем ва дар охир пас аз тамомшавии давр тағйирёбандаи Summa – ро ба ҷоп медиҳем, ки дар он ҳамаи элементҳо гирд оварда шудаанд. Мисол:

```

#include <iostream>
using namespace std;
int main(){
    const int n=5;
    int summa=0;
    int a[]={1,3,5,7,9};
    for (int i=0;i<n;i++)
        summa+=a[i];
    cout<<summa<<endl;
    return 0;
}

```

Ин барномаро метавон бо истифода аз оператори *коммо* ба намуди зерин низ навишт. **Мисол:**

```

#include <iostream>
#include <cmath>
using namespace std;
int main(){
    const int n=5;
    int summa,i;
    int a[]={1,3,5,7,9};
    for (summa=0, i=0;i<n;summa+=a[i],i++);
    cout<<summa<<endl;
    return 0;
}

```

Барои зарб намудани элементҳои массив дар аввал тағйирёбандае муаррифӣ мекунем ва ба он ба сифати қимати аввала 1 мебахшем ва баъд ба ёрии оператори даврӣ ҳамаи элементҳои массивро ба қимати тағйирёбандаи мазкур зарб мекунем, дар натиҷа ҳосили зарби элементҳои массив ба даст меоянд. **Мисол:**

```

#include <iostream>
using namespace std;
int main(){
    const int n=5;
    long int Z;
    int i;
    int a[]={-2,1,4,7,-6};
    for (Z=1, i=0;i<n;Z*=a[i],i++);
    cout<<"Z="<<Z<<endl;
    return 0;
}

```

Ба намуди диаграмма тасвир намудани элементҳои массив

Элементҳои массивро метавонем ба намуди график, яъне гистограмма низ тасвир кард. Вобаста ба миқдори ҳар як элементи массив дар саҳифаи тасвир ягон рамз чоп карда мешавад.

```

#include <iostream>
#include <iomanip>
using namespace std;
int main(){
    const int n=5;
    int i,j;
    int a[]={20,8,25,17,10};
    cout<<"Element"<<setw(13)<<"Qimat" <<setw(17)<<"histogramma"<<endl;
    for (i=0;i<n;i++){
        cout<<setw(7)<<i<<setw(13)<<a[i]<<setw(9);
        for(j=0;j<a[i];j++)
            cout<<"*";
        cout<<endl;
    }
    return 0;
}

```


Тавре ки дида мешавад, дар ин мисол функсияи `setw ()` аз библиотекаи «`iomanip.h`» истифода бурда шудааст. Функсияи мазкур барои пур намудани саҳифаи тасвир ба ҷои холӣ истифода бурда мешавад, ки дар оянда дар борааш пурратар эзоҳ медиҳем. Ду ҳалкаи такрории дигар, ки якеаш барои элементҳои массивро дар саҳифаи тасвир ҷоп намудан ва дигараш бошад, ба миқдори аз 0 то ҳар як элементҳои массив дар саҳифаи тасвир рамзи ситорача «*» ҷоп мекунад. Натиҷаи барнома ба сурати зерин аст:

Element	Qimat	histogramma
0	20	*****
1	8	*****
2	25	*****
3	17	*****
4	10	*****

Бо ададҳои тасодуфӣ пур намудани элементҳои массив

Агар массив барои ягон масъалаи санҷишӣ истифода шуда бошад, пас, метавон элементҳои онро аз ададҳои тасодуфӣ пур кард. Барои иҷрои ин амал аз функсияи `rand()` ва ё `randomize()` истифода бурда мешавад. Дар функсияҳои мазкур метавонем ҳудуди болоии массивро нишон диҳем. Масалан, агар талаб карда шавад, ки элементҳои массивро аз ададҳои тасодуфӣ аз 50 калон набуда пур кунед, барои ин яке аз операторҳои зерин истифода бурда мешаванд:

$A[i]=rand()\%50$ ё $A[i]=random(50)$.

Барои массивро аз ададҳои манфӣ пур кардан бошад, ин операторҳо ба сурати зерин истифода бурда мешаванд:

$A[i]=rand()\%50-55$ ё $A[i]=random(50)-55$.

Агар ҳудуди болоӣ дар массив нишон дода нашуда бошад, компилятор ба таври ихтиёрӣ аз ададҳои бузург массивро пур мекунад.

Мисол. Барномае нависед, ки дар он элементҳои массив ба таври тасодуфӣ интихоб карда шуда бошанд ва дар саҳифаи тасвир нишон дода шаванд.

```
#include <iostream>
using namespace std;
#define n 10
int main()
{
    int a[n];
    for (int i=0;i<n;i++){
        a[i]=rand()%45;
        cout<<a[i]<<" ";
    }
    cout<<endl;
    return 0;
}
```

Тавре ки мебинед, дар ин чо аввал андозаи массив ҳамчун адади доимӣ тавассути роҳнамуни #define муаррифӣ карда шудааст. Дар дохили функсияи асосӣ массиве бо андозаи n муаррифӣгардида, дар дохили ҳалқаи такрорӣ аз ададҳои тасодуфӣ пур карда шудааст. Дар дохили ҳалқаи мазкур элементҳои массив дар саҳифаи тасвир ба чоп омода карда шудаанд. Натиҷаи барномаи мазкур намуди зеринро дорад.

40	2	10	1	7	0	25	43	31	4
----	---	----	---	---	---	----	----	----	---

Чустуҷӯи элементи калонтарин ва рақами тартибии он дар массив

Масъала: Массиви a дорои n элемент мебошад. Барномае нависед, ки элементи калонтарин ва рақами тартибии онро муайян кунад.

Алгоритми ин масъала чунин аст: дар аввал ду тағйирёбандаро ба номи тах ва nтах муаррифӣ мекунем ва фарз мекунем, ки элементи аввала калонтарин аст. Элементи мазкурро ба тағйирёбандаи тах ва рақами тартибии онро ба тағйирёбандаи nтах мебахшем. Баъд аз он тах ба элементҳои ояндаи массив муқоиса карда мешавад, агар тах аз он бузургтар бошад, ҷойи уро ба тах иваз мекунем ва агар тах аз он элемент хурд бошад, элементи мазкурро ба тах мебахшем, яъне ба ҷои он элементи тах навишта мешавад ва рақами тартибии он ба nтах бахшида мешавад. Ин амалро ба элементҳои боқимондаи массив низ иҷро мекунем. Барои пурратар фаҳмо шудани ин амал онро дар ҷадвали зерин гирд овардем.

№	0	1	2	3	4	5	6	7	8	9
Массиви додашуда	5	7	9	0	3	19	5	8	21	13
Қимати тах	5	7	9	9	9	19	19	19	21	21
Қимати nтах	1	2	3	3	3	5	5	5	8	8

Барномаи алгоритми ёфтани элементи калонтарин.

```
#include <iostream>
using namespace std;
int main()
{
    const int n=10;
    int i,j,max,nmax,a[n];
    for (i=0;i<n;i++)
    {
        a[i]=rand()%25;
        cout<<"a["<<i<<"]="<<a[i]<<endl;
    }
    for (max=a[0],nmax=0,i=0;i<n;i++)
        if (max<a[i])
        {
            max=a[i];
            nmax=i;
        }
    cout<<"\nmax="<<max<<" \n ";
    cout<<"nmax="<<nmax<<" \n ";
    return 0;
}
```

Тавре ки дида мешавад, дар ин чо элементҳои массив аз ададҳои тасодуфӣ аз 25 калон набуда, пур карда шудаанд, сипас барнома аз рӯи алгоритми дар боло овардашуда амал намуда, элементҳои калонтарин ва рақами тартибии онро мебарорад.

Равон кардани массив ба функция ба сифати аргумент

Тавре ки қаблан гуфтем, дар забони C++ ҳангоми ба функция равон кардани аргумент ду роҳ мавҷуд аст: якум, ба ёрии адрес (суроға), дуюм, ба ёрии қимат. Элементҳои массивро низ метавонем бо ёрии ин ду роҳ ба функция равон кунем. Дар забони C++ номи массив ҳамчун суроғаи аввалини массив фаҳмида мешавад. Ҳангоми ба функция равон кардани массив номи массив бе истифодаи қавси фигуравӣ навишта мешавад.

Ҳангоми массивҳоро ба сифати аргумент ба функция равон кардан номи массив дар функция бо қавсҳои фигуравӣ навишта шуда, андозаи массив бошад, берун аз қавси фигуравӣ, баъди оператори *комма* навишта мешавад. Ҳангоми даъват намудани функция дар функцияи аслии номи массив бе қавси фигуравӣ ва андозаи массив навишта мешавад. Ин навъ равонкун монанди тағйирёбандаҳои оддӣ фурӯзонӣ ба миқдор (call by value) номида шуда, дар он нусхаи ҳамаи элементҳо ба функция равон карда мешаванд. Мисол :

```
#include <iostream>
#include <iomanip>
using namespace std;
void modifyArray(int[],int);
int main(){
    const int n=10;
    int a[n],i;
    for (i=0;i<n;i++){
        a[i]=rand()%30;
        cout<<setw(3)<<a[i];
    }
    cout<<endl;
    modifyArray(a,n);
    for (i=0;i<n;i++){
        cout<<setw(3)<<a[i];
    }
    return 0;
}

void modifyArray(int a[],int n)
{
    for (int i=0;i<n;i++)
        a[i]*=2;
}
```

Тавре ки дар ин чо дида мешавад, дар пешнамунаҳои функсия ва худӣ функция номи массив бе истифодаи андозаи он (андозаи он берун аз маҷмаи муқаррариаш пас аз оператори *комма*) навишта шудааст, яъне `void modifyArray(int[],int)`. Ҳангоми фурӯзонӣ бошад, танҳо ном ва андозаи массив

бе истифодаи қавсҳои фигуравӣ ба намуди зерин навишта шудааст:
modifyArray(a,n);

Роҳи дигари раво кардани массив ба функсия. Дар функсия танҳо суроғи массив нишон дода мешаваду халос. Мисоли гузаштаре метавон ба намуди зерин низ навишт.

```
#include <iostream>
#include <iomanip>
using namespace std;

void arrayprint (int *a,int n){
    int i;
    for(i=0;i<n;i++)
        cout<<a[i]<<" ";
}
int main()
{
    const int n=10;
    int a[n],i;
    for (i=0;i<n;i++)
        a[i]=rand()%30;
    arrayprint(a,n);
    return 0;
}
```

Дар мисоли мавриди назарамон ба функсияи arrayprint (int *a,int n) суроғи массив нишон дода шудааст, яъне *a ишорагар ба массив мебошад. Дар оянда ин матлабро пурратар мавриди баҳс қарор медиҳем.

Массивҳои динамикӣ

Ба намуди пешфарз ҳангоми муаррифӣ намудани массив дар хотира вобаста ба андозаи он маҳал (ҷой)-е ҷудо карда мешавад. Вале дар баъзе масъалаҳои андозаи массив пешакӣ маълум набуда, дар баъзеи дигарашон ҳангоми иҷро намудани барнома зарурати андозаи массивро тағйир додан пеш меояд, барои ин лозим меояд, ки хотираи динамикиро истифода барем. Ҳангоми массивро аз навъи динамики муаррифӣ кардан дар аввал массив ҳамчун як ишорагар муаррифӣ карда мешавад. Баъд аз дохил намудани андозаи массив барои он ҳофиза ҷудо карда мешавад.

Ҳангоми ба намуди статикӣ (қарордоди) муаррифӣ намудани массив барои нигоҳ доштани элементҳои массив як блоке аз ҳофиза ҷудо карда мешавад. Агар массив ҳамчун як ишорагар муаррифӣ гардад, он гоҳ, барои он чунин ҷой ҷудо (reserve) карда намешавад. Дар натиҷа барои ишорагарро ҳамчун як массив истифода бурдан зарур аст, ки ҳофизае барои элементҳои массив ҷудо карда шавад. Барои ҷудо намудани чунин намуди ҳофиза яке аз роҳҳои зерин истифода бурда мешавад:

- 1) Бо истифода аз функсияҳои malloc , calloc, realloc ва free
- 2) Бо истифода аз операторҳои new ва delete

Функция `malloc` як блок ҳофизаро барои нигоҳ доштани объекти нишондодашуда `чудо (reserve)` намуда, ба катакчаи якуми он блок ишорагар бармегардонад. Наҳви истифодабарии ин функция намуди зеринро дорад:

```
void *malloc(size)
```

Дар ин ҷо `size` -андозаи ҳофизаи `чудо` ба ҳисоби байт ифода мекунад. Агар `чудо` намудани ҳофиза бомуваффақият анҷом пазирад, он гоҳ ин тобеъ тағйирёбандаи навъи `*void`-ро бармегардонад. Дар ҳолати акс бошад, қимати `Null` бармегардонад.

Мисол. Бигзор, мо мехоҳем массиве аз навъи бутун бо номи `a`, ки дорои `n` элемент мебошад, эълон кунем. Дар аввал онро ҳамчун ишорагари ба сурати зерин муаррифӣ мекунем: `int *a;`

Дар ин равиш ҳофиза `чудо` карда намешавад, пас аз дохил кардани `n` барои `чудо` намудани ҳофиза аз функцияи `malloc` ба сурати зерин истифода мебарем:

```
a=malloc (n*sizeof(int));
```

Ин оператор як блок ҳофизаро барои нигоҳ доштани 15 элементи массив `чудо` мекунад. Дар ин ҷо оператори `sizeof()` андозаи навъи `int`-ро бо ҳисоби байт баргардонида, онро ба миқдори элементҳои массив зарб мезанад. Аз ин ҷо ҳофизаи мавриди назар `чудо (reserve)` карда мешавад. Тағйирёбандаи `a` аз навъи бутун муаррифӣ карда шудааст ва тобеъи `malloc` бошад, ишорагари рамзӣ бармегардонад. Аз сабаби он ки дар забони C/C++ рамзҳо низ ҳамчун адади бутун фаҳмида мешаванд, пас, оператори дар боло истифодабурдамон қобили қабул аст.

Барои бо боварии комил истифода бурдани тобеъи мазкур хубтар аст, ки дар дохили он амали табдилдиҳии навъ (`cast`)-ро низ истифода барем. Аз ин ҷо оператори дар боло навиштаамон намуди зеринро мегирад:

```
a=(int *) malloc (n*sizeof(int));
```

Функцияи `calloc` низ барои `чудо` намудани ҳофизаи динамикӣ истифода бурда мешавад. Наҳви умумии он намуди зеринро дорад:

```
void * calloc (num, size)
```

Функцияи мазкур барои ба массиви `num` ба андозаи `size` `чудо` намудани ҳофиза хизмат мекунад. Агар амали `чудо` намудани ҳофиза бо муваффақият анҷом гирад, он гоҳ ин функция ба мавқеи `чудо` кардашудаи ҳофиза ишорагар бармегардонад. Дар акси ҳол бошад, қимати `Null` бармегардонад.

Функцияи `realloc` бошад, андозаи ҳофизаи пештар `чудо` кардашударо метавонад тағйир диҳад. Наҳви умумии ин функция намуди зеринро дорад:

```
char *realloc (void *p, size)
```

Дар ин ҷо `p` -ишорагар ба мавқеи ҳофизаест, ки бояд ба андозаи `size` тағйир дода шавад. Агар ҳангоми иҷроиши функцияи мазкур суроғаи мавқеи ҳофиза тағйир дода шавад, он гоҳ суроғаи нав ба сифати натиҷа баргардонида мешавад. Агар қимати параметри якум `Null` бошад, он гоҳ ин функция монанди функцияи `malloc` кор карда, ба андозаи `size` ҳофизаро `чудо` мекунад.

Барои озод намудани ҳофизаи `чудо` кардашуда аз функцияи `free()` истифода бурда мешавад. Наҳви умумии ин функция намуди зеринро дорад:

```
void free (void *p)
```

Дар ин чо p ба он мавқеи ҳофизае ишорат мекунад, ки пештар тавассути функсияҳои `malloc`, `calloc` ва ё `realloc` чудо карда шудааст.

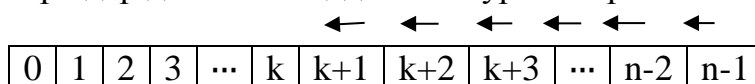
Роҳи дигари чудо кардани ҳофизаи динамикӣ истифода аз оператори `new` мебошад. Барои аз ин оператор истифода бурдан дар аввал массив ҳамчун як ишорагар муаррифӣ карда шуда, сипас андозаи он бо ёрии оператори `new` муайян карда мешавад. **Мисол:**

```
int n, *a;
....
a=new int[n];
```

Дар ин чо тавре ки дида мешавад, дар сатри якум ду тағйирёбандаи бутун, ки якеаш аз навъи ишорагар буда, номи массивро ифода мекунад, муаррифӣ карда шудааст. Баъдан бо ёрии оператори `new` андозаи тағйирёбандаи ишорагар нишон дода шудааст. Андозаи массивро ҳангоми иҷроиши барнома аз берун низ дохил кардан мумкин аст. Барои озод намудани хотира бошад, аз оператори `delete []` истифода бурда мешавад.

Илова ва нест намудани элемент дар массив

Бигзор массиве дорои n –элемент дода шудааст. Талаб карда мешавад, ки элементи k -уми он нест карда шавад. Бигзор, k рақами тартибии ягон элементи бошад, он гоҳ элементҳои массив чунин тағйир меёбанд: элементи $k+1$ -ум ҷойи элементи k -ум ва элементи $k+m$ -ум ҷойи элементи $k+m-1$ -умро ишғол мекунад. Барои хубтар шинос шудан ва дарк намудани ин матлаб, ҳамаи элементҳои массивро дар дохили як чадвал ба сурати зерин тасаввур мекунем:



Бо суҳанҳои дигар элементи $k-1$ -ум сарфи назар шуда, аз элемент k -ум сар карда, ҳар як элемент як мавқеъ ба тарафи чап кӯчонида мешаванд. Дар натиҷа элементҳои массив $n-1$ то мешаванд.

Матни барнома:

```
#include <iostream>
using namespace std;
int main()
{
    int i,k;
    int n,*a;
    cout<<"n=";cin>>n;
    cout<<"k=";cin>>k;
    a=new int [n];
    for (i=0;i<n;i++){
        a[i]=rand()%45;
        cout<<"a["<<i<<"]="<<a[i]<<endl;
    }
    cout<<endl;
    for (i=k;i<n-1;i++)
        a[i]=a[i+1];
    n--;
    for (i=0;i<n;i++)
```

```

        cout<<"a["<<i<<"]="<<a[i]<<endl;
    delete []a;
    return 0;
}

```

Барои нест кардани элементе ё элементҳо аз массив, метавон аз функсияҳо низ истифода бурд.

Мисоли 1. Барномае нависед, ки нахустин элементи манфии массиви додашуда нест кунад.

```

#include <iostream>
using namespace std;
const int n=10;
int a[n];
void shiftLeft(int pos, int m, int dsp, int *a)
{
    for(int i=pos; i<n-dsp;i++)
        a[i]=a[i+dsp];
    m-=dsp;
}
int main()
{
    int i, j=0;
    int a[n];
    cout<<"massivi dodashuda:\n";
    for (i=0;i<n;i++){
        a[i]=rand()%45-5;
        cout<<a[i]<<" ";
    }
    cout<<"\nmassiv pas az nest kardani elementho:"<<endl;
    while(j<n && a[j]>0)
        j++;
    if (j<n)
        shiftLeft(j,n,1,a);
    for (i=0;i<n-1;i++)
        cout<<a[i]<<" ";
    cout<<endl;
    return 0;
}

```

Дар ин ҷо аргументи якуми функсия мавқеъ ё рақами тартибии элементро, ки бояд аз массив нест карда шавад, ифода намуда, аргументи дуввум андозаи массивро нишон медиҳад. Аргументи саввум миқдори элементҳои нестшавандаро нишон медиҳад ва аргументи охирон ишорагар ба массив мебошад. Дар функсияи аслии ҳалқаи такрорӣ while() барои дастрас шудан ба рақами тартибии элементи мавриди назар истифода шудааст. Натиҷаи барнома шакли зеринро дорад:

```

massivi dodashuda:
35 -3 5 -4 2 -5 20 38 26 1
massiv pas az nest kardani elementho:

```

35 5 -4 2 -5 20 38 26 1

Ба ҳамин монанд функсияе барои ба массив гузоштани элемент метавон таъриф кард.

Мисоли 2. Барномае нависед, ки пас аз элементи якуми мусбати массив адади 100-ро гузорад.

```
#include <iostream>
using namespace std;
const int n=10;
int a[n];
void shiftRight(int pos, int m, int dsp, int *a){
    for(int i=m; i>=pos;i--){
        a[i+dsp]=a[i];
        m+=dsp;
    }
}
int main(){
    int i, j=0;
    int a[n];
    cout<<"massivi dodashuda:\n";
    for (i=0;i<n;i++){
        a[i]=rand()%50-20;
        cout<<a[i]<<" ";
    }
    cout<<"\nmassiv pas az nest kardani elementho:"<<endl;
    int k=0;
    while((k<n) && (a[k]>=0))
        k++;
    if (k<n) { j=k;
    shiftRight(k,n,1,a);
    a[j+1]=100;
    }
    for (i=0;i<=n;i++)
        cout<<a[i]<<" ";
    cout<<endl;
    return 0;
}
```

Ин барнома низ монанди барномае пешина буда, фақат баръакси он амал мекунад.

Массивҳо аз сатрҳои матнӣ

Мо то ҳоло ба массивҳое, ки элементҳои он аз навъи шуморишӣ (ададӣ) будаанд, сарукар гирифтаем. Илова бар ин, элементҳои массив метавонанд дилхоҳ навъи стандартӣ бошанд. Ҳоло массивҳоеро дида мебароем, ки элементҳои он сатрҳо (риштаҳо) мебошанд.

Дар забони C++ сатрҳо ҳамчун массивҳои рамзӣ шинохта шуда, охири ҳар сатр бо рамзи Null (охири сатр) ба поён мерасад. Дар дохили барнома рамзи мазкур дар қолаб (форма)-и escape square бо '\0' нишон дода мешавад. Индекси поёнии элементҳои як массив аз навъи сатрҳо 0 буда, индекси болоӣ

якто кам аз микдори рамзҳои ҳамон сатр мебошад. Масалан, сатри «Hello world» як массиви рамзии дорои 11 рамз мебошад. Дар ин навъ массивҳо ҷойи ҳолӣ низ ҳамчун як элемент фаҳмида мешавад. Ба массивҳои сатрӣ низ дар вақти муаррифӣ қардан метавон қимати аввала бахшид. Масалан:

```
char Arsatr[]="Hello World";
```

Массиви муаррифӣшуда дорои 11 элемент мебошад. Инчунин массивҳои сатрӣро метавонанд ҳар як элементашон алоҳида бошанд. Масалан: массиви болоро метавонем ба сурати зерин низ таъриф кард:

```
char Arsatr[]={ 'H','e','l','l','o','\0','W','o','r','l','d' }
```

Тавре ки дида мешавад, дар ин ҷо ҳар як элементи массиви рамзӣ дар дохили ноҳунаки якто-якто навишта шуда, барои ҷойи ҳолӣ (фосила) рамзи '\0' истифода шудааст ва оператори коммо барои аз ҳамдигар ҷудо намудани рамзҳо ба кор рафтааст. Барои дар ин навъ массивҳо ба элементи зарурӣ дастрас шудан, аз номи массив ва зернависи (рақами тартиби)-и элементи мавриди назар истифода бурда мешавад.

Монанди дигар массивҳо зернависи элементҳои массиви рамзӣ низ аз нол оғоз ёфта, ба n-1 ба охир мерасад, яъне дар массиви мазкур элементи нолум ин рамзи (харфи) «H» буда элементи n-1 (11-1)-ум рамзи «d» мебошад.

Мисол. Барномае нависед, ки ҳамаи элементҳои массиви Arsatr[]-ро ба ҷоп диҳад. Инчунин элементи аввалин ва охиринашро низ ба ҷоп диҳад. Мисол:

```
#include <iostream>
using namespace std;
int main(){
    const int n=11;
    cout<<"Massivi dodoshuda=";
    char Arsatr[]={ 'H','e','l','l','o','\0','W','o','r','l','d' };
    for (int i=0;i<n;i++)
        cout<<Arsatr[i];
    cout<<endl;
    cout<<"Elementi yakum="<<Arsatr[0]<<"\nElementi duyum="<<
        Arsatr[1]<<endl;
    return 0;
}
```

Чи хеле ки мебинем, дар ин ҷо аввал доимие муаррифӣ шуда аст, пас аз он массив бо элементҳои аввалааш таъриф шудааст. Ҳалқай такрорӣ то замони элементҳои массив ба охир расидан давр зада, дар ҳар итератсия як элементи массивро бо ёрии оператори хоричкунӣ ба саҳифаи тасвир раво мекунад (менависад).

Тавре ки гуфтем, дар оператори хоричкунӣ ифодаи Arsatr[0] ва ифодаи Arsatr[n-1] нишондиҳандаи элементҳои аввалин ва охири массив буда, элементҳои мазкурро ба саҳифаи тасвир пешниҳод мекунанд. Натиҷаи барнома шакли зеринро дорад:

```
Massivi dodoshuda=
Elementi yakum=H
Elementi duyum=e
```

Дар вақти муаррификунӣ метавонем танҳо андозаи массивро нишон дода, элементҳои онро аз клавиатура дохил кунем. Масалан, бигзор массиве бо номи name [] ки андозааш ба 15 баробар аст, ба сурати зерин муаррифӣ карда шуда бошад: char name[15];

Барои элементҳои массиви мазкурро тавассути дастури дохилкунии cin аз клавиатура дохил намудан андозаи массив навишта намешавад.

Мисол. Барномае нависед, ки дар он номи як нафар донишҷӯӣ аввал аз клавиатура дохил карда шуда, Сипас ба чоп дода шавад.

```
#include <iostream>
using namespace std;
int main(){
    const int n=15;
    cout<<"Enter you name\n";
    char name[n]={0};
    cin>>name;
    for (int i=0;i<n;i++)
        cout<<name[i];
    cout<<endl;
    return 0;
}
```

Ҳангоми аз клавиатури ворид кардани массиви рамзӣ объекти cin то расидан ба рамзи фосила сатри додашударо мехонаду халос. Дар оянда дар ин бора эзоҳи пурратар медиҳем.

Мураттабсозӣ ва ҷустуҷӯии элементҳо дар массив

Мураттабсозӣ ва ҷустуҷӯии элементе дар дохили як маҷмӯъ дар аксар ҷойҳо, ба хусус дар системаҳои иттилоотии ҷустуҷӯӣ, тичоратӣ, ёфтани луғат аз фарҳанг, ёфтани китоб аз китобхона ва ғайра корбурди зиёде дорад. Мақсади асосии мураттабсозӣ (sorting) ба тартиб аз рӯи қоидаи муайяне (аз рӯи алифбо, ё зиёдшавии қимат) гузоштани додаҳо (дар массив элементҳо) мебошад, ки он ҷустуҷӯии элементҳои заруриро осон мекунад. Чунончи агар додаҳо аз аз рӯи қоидаи муайяне дастабандӣ нашуда бошанд, ёфтани як унсури муайяне дар дохили додаҳои душвор ҳатто имконнопаз мегардад. Биёед, фарҳанг (dictionary)-ро ки дар он луғатҳои зиёде мавҷуданд, пеши назар меорем. Агар дар фарҳанг калимаҳо аз рӯи алифбо ҷобачо нашуда бошанд, барои ҷустуҷӯии калимаи мавриди ниёз чи қадар вақт беҳуда сарф мешавад, яъне барои ёфтани он калима зарур мешавад, ки ҳамаи саҳифаҳои фарҳангро аз назар гузаронем. Пас, барои сарътар ва осонтар дастрас шудан ба калимаи мавриди ниёз агар дар фарҳанг калимаҳо аз рӯи алифбо ҷобачо шуда бошанд, танҳо саҳифаи лозимиро мебинему бас. Пас, биёед аввал алгоритмҳои мураттабсозӣ ва баъд алгоритмҳои ҷустуҷӯиро дида мебароем.

Гурӯҳбандии равишҳои мураттабкунӣ

Алгоритмҳои мураттабсозии элементҳои массив ба ду гурӯҳ ҷудо карда мешаванд:

1) Мураттабсозии дохилӣ (internal sort)-ки дар он ҳамаи додаҳо дар хотираи компютер қарор дода мешаванд.

2) Мураттабсозии хориҷӣ (external sort), ки дар он додаҳои мавриди назар дар ҳофизаҳои бегона (аҷнаби) монанди навор ё саҳифаҳо қарор доранд, ки барои ин гуна додаҳо истилоҳи файл (File) ба кор бурда мешавад, ки он дар мавзӯҳои оянда мавриди баҳс қарор мегирад.

Мураттабсозии дохилӣ аз рӯи алгоритм ё равишҳои зерин амали карда мешавад:

- 1) bubble sort
- 2) shell sort
- 3) tosh sort
- 4) selection sort
- 5) insertion sort
- 6) quick sort
- 7) merge sort
- 8) binary tree sort
- 9) heap sort
- 10) radix sort (bucket sort)

Мураттабсозии ҳубобӣ (bubble sort)

Яке аз содатарин равишҳои мураттабсозӣ мураттабсозии ҳубо-бӣ мебошад, агарчанде ки дар муқоиса нисбат ба равишҳои дигар ҷойи поёнтареро ишғол мекунад, вале хело сода буда, ба истифодабаранда фаҳмо мебошад. Вожаи bubble мафҳуми илмӣ набуда, танҳо дар луғат ба маънои ҷушидан ва ғулғула кардан омадааст. Вақте ки зарфи обро ҷӯшонидем, ҳубобчаҳои ҳаво, ки дар об мавҷуданд, дар сатҳи об ҳаракат мекунанд. Дар равиши мазкур элементҳои калонтарин ба ҳубобчаҳои зарфи об монанд ҳар маротиба ба самти боло ё интиҳо ҳаракат мекунад. Аз ин ҷост, ки ин навъ мураттабсозиро мураттабсозии ҳубобӣ номидаанд.

Дар равиши мураттабсозии мазкур элементҳои якум ба элементҳои дуввум муқоиса карда мешаванд, агар узви якум аз дуҷум калонтар бошад, ҷойи онҳо иваз карда мешавад. Сипас, ин амал дар мавриди элементҳои дуҷум бар сеҷум ва ҳамин тариқ барои элементҳои боқимонда сурат мегирад, яъне дар охир элементҳои $n-1$ -ум бо элементҳои n -ум муқоиса карда мешавад, агар элементҳои $n-1$ – ум аз элементҳои n -ум калон бошад, ҷояшон иваз карда мешавад. Вақте ки ин амал як маротиба иҷро карда шуд, элементҳои калонтарин дар охир қарор мегирад. Акнун фарз мекунем, ки элементҳои n -уми массив вучуд надорад. Амали пештар иҷро кардамонро ба $(n-1)$ элементҳои боқимонда якбор иҷро карда, элементҳои калонтаринро дар ҷойи $(n-1)$ -ум қарор медиҳем. Агар ин амалро $(n-1)$ маротиба такрор карда, ҳар маротиба як хонаро номида гирем, дар охир элементҳои массив ба тартиби зиёдшавиашон ба намуди зерин ҷобачо карда мешаванд:

$$a[0] \leq a[1] \leq a[2] \leq \dots \leq a[n-1]$$

Агар хоҳем, ки элементҳои массив бо тартиби камшавиашон (отсортировать) ҷойгир карда шаванд. Ҳангоми муқоисакунӣ ҷойи элементҳои

хурдтаринро аз мавқеи $a[i-1]$ ба $a[i]$ баръакси равиши боло иваз мекунем. Дар ин сурат элементҳои массив ба сурати зерин ҷобачо карда мешаванд.

$$a[0] \geq a[1] \geq a[2] \geq \dots \geq a[n-1]$$

Мисол. Барномае нависед, ки элементҳои массиви додашударо аз рӯи равиши ҳубоби бо тартиби зиёдшавиашон ҷобачо гузорад.

```
#include <iostream>
using namespace std;
int main()
{
    const int n=25;
    int a[n];
    int i,b,j;
    cout<<" Elementhoi massiv pesh az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)
    {
        a[i]=rand()%45;
        cout<<a[i]<<" ";
    }
    cout<<endl;
    for (j=1;j<n;++j)
        for (i=0;i<n-j;++i)
            if (a[i]>a[i+1])
            {
                b=a[i];
                a[i]=a[i+1];
                a[i+1]=b;
            }
        cout<<endl;
        cout<<" Elementhoi massiv pas az
jobajoguzori"<<endl<<endl ;
        for (i=0;i<n;i++)
            cout<<a[i]<<" ";
        cout<<endl;
        return 0;
}
```

Дар ин ҷо аввал массиве аз навъи бутун, ки дорои n ($n=25$) элемент мебошад, муаррифӣ карда шудааст. Баъд элементҳои массив аз ададҳои тасодуфи пур карда шуда, бо ёриии оператори хориҷкуни ба хориҷ карда шудаанд. Сипас ду давр истифода шудааст, даври якум шароит фароҳам меоварад, ки $(n-1)$ -маротиба барои ҳар як элементи массив амали муқоисакуни иҷро карда шавад. Дар аввал элементи якум (бо индекси 0) бо элементи дуввум муқоиса карда мешавад. Агар он аз элементи дуввум калон бошад, қимати он ба тағйирёбандаи ёрирасони b бахшида мешавад. Сипас элементи $a[i]$ ба элементи $a[i+1]$ иваз карда мешавад, ки дар натиҷа тағйирёбандаи b аз қимат холи мешавад ва элементи $a[i+1]$ бошад, қимати худро ба тағйирёбандаи b мебахшад ва ҳоказо. Тавре ки дар боло гуфтем, элементҳои массив ҷобачо карда мешаванд. Баъди иҷрои амалҳои мазкур элементҳои массив бори дигар ба сурати зерин хориҷ карда мешаванд.

Elementhoi massiv pesh az jobajoguzori

40 2 10 1 7 0 25 43 31 4 23 16 19 42 10 12 36 8 27 29 31 35 35 29 14

Elementhoi massiv pas az jobajoguzori

0 1 2 4 7 8 10 10 12 14 16 19 23 25 27 29 29 31 31 35 35 36 40 42 43

Мураттабсозӣ бо равиши интиҳоб (selection sort)

Моҳияти ин навъ мураттабсозӣ аз он иборат аст, ки дар миёни элементҳои массив элементи калонтарин ҷустуҷӯ карда шуда, ҷойи он ба элементи охири иваз карда мешавад. Баъд аз он фарз карда мешавад, ки элементи n -ум вучуд надорад ва аз байни $(n-1)$ -элементи боқимонда калонтаринаш ҷустуҷӯ карда мешавад, ҳангоми ба даст омадани он ҷояш бо элементи $(n-1)$ -ум иваз карда мешавад. Амали мазкур $n-1$ маротиба такрор карда мешавад. Дар охир танҳо ду элемент яқум ва дуум боқӣ мемонанд, ки аз онҳо калонтаринашро ба хо-наи дуввум ҷой дода мешавад. Ба ҳамин тариқ, элементҳои массив ҷо-баҷо карда мешаванд.

Бартарии ин равиш назар ба равиши ҳубобӣ аз он аст, ки дар ин навъ мураттабсозӣ ҷойивазкунии элементҳо камтар ба амал меояд, зеро дар ин равиш дар ҳар такрор фақат як маротиба ҷойивазкунӣ ба амал меояд.

Мисол: Барномаи усули ҷобачогузори интиҳоб.

```
#include <iostream>
#include <iomanip>
using namespace std;
int main(){
    const int n=25;
    int i,j,max,b,name;
    int a[n];
    cout<<"Elementhoi massiv pesh az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++){
        a[i]=rand()%80;
        cout<<a[i]<<" ";
    }
    cout<<endl;
    for (j=1;j<n;b=a[n-j],a[n-j]=a[name],a[name]=b,j++){
        for (max=a[0],name=0,i=1;i<=n-j;i++)
            if(a[i]>max)
            {
                max=a[i];
                name=i;
            }
        cout<<endl;
        cout<<" Elementhoi massiv pas az
jobajoguzori"<<endl<<endl ;
        for (i=0;i<n;i++)
            cout<<a[i]<<" ";
        cout<<endl;
        return 0;
    }
}
```

Мураттабсозии гузориши (insertion sort)

Бигзор массиви a дорои n элемент бошад, яъне $a_1, a_2, a_3, \dots, a_n$. Барои ба тартиб гузоштани элементҳои массиви мазкур дар аввал ду элементро мегирем, яъне a_1 ва a_2 ва фарз мекунем, ки онҳо мурат-табанд, баъд аз он, аъзои сеюм, яъне a_3 -ро интихоб карда, онро нисбат ба ду аъзои аввала дар ҷойи муносибе мегузorem, ки маҷмӯи дорои се элемент мураттаб ҳосил гардад. Ба ҳамин монанд то элементҳои n -ум давом дода мешавад, ки дар охир ҳамаи элементҳои массив ҷобачо карда мешаванд. Барои ин матлабро пурратар дида баромадан биёед массиви дорои 4-элементро дида мебароем. Мисол:

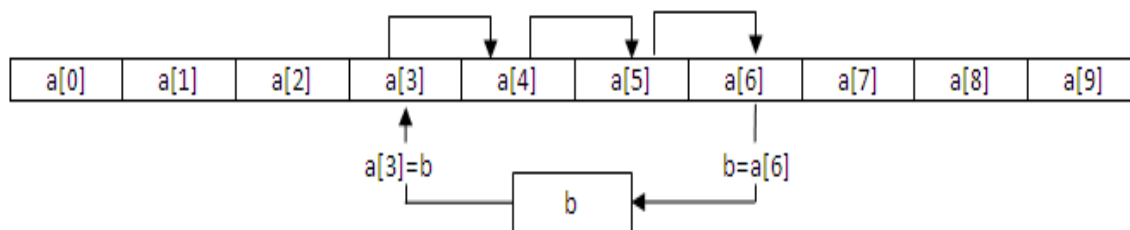
Элементҳои массиви дода шуда: $\rightarrow 4, 3, 1, 2$;

Қадами 1: $\rightarrow 3, 4, 1, 2$;

Қадами 2: $\rightarrow 1, 3, 4, 2$;

Қадами 3: $\rightarrow 1, 2, 3, 4$;

Тавре ки дида мешавад, барои мураттабсозии чор элементҳои массив 3-итератсия истифода карда шудааст. Барои пурратар фаҳмо шудан, массиви дорои 10 элементро пеши назар меорем ва фарз мекунем, ки 6 элементҳои он мураттаб шуда, элементҳои шашуми аз элементҳои чорум хурд (ё баробар) аст. Ба тағйирёбандаи иловагӣ элементҳои шашумро бахшида, баъд онро дар ҷойи чорум қарор медиҳем. Ба ҳамин монанд ҳар як элементҳои номураттаби массив ба элементҳои ҷойи муносиб иваз карда мешаванд. Дар ҷадвали поён ин равиш оварда шудааст.



Мисол: Барномаи усули мураттабкунии гузориш навишта шавад.

```
#include <iostream>
using namespace std;
int main(){
    const int n=25;
    int i,j,max,b,name;
    int a[n];
    cout<<" Elementhoi massiv pesh az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)    {
        a[i]=rand()%100;
        cout<<a[i]<<" ";
    }
    cout<<endl<<endl;
    for (i=1;i<n;a[j+1]=b,i++)
        for (b=a[i],j=i-1;(j>-1 && b<a[j]);a[j+1]=a[j],j--);
    cout<<" Elementhoi massiv pas az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)
        cout<<a[i]<<" ";
    cout<<endl;
    return 0;
}
```

Мураттабсозии tag (tag sort)

Чанде қабл мо алгоритми мураттабсозиеро ба номи мурат-табсозии хубобӣ дида баромада будем. Акнун фарз мекунем, ки мас-сиви дорои андозаи бузург дода шуда бошад. Масалан, дорои 1млн. элемент барои ҳар як элементро муқоиса намудан вақти зиёд сарф мешавад. Барои ин навъ массивҳо ва массивҳоеро ки элементҳои онҳо сатрҳо, рамзҳо ва сохторҳо мебошанд, равиши нави мураттабсозиро ба намуди tag sort истифода мешавад. Моҳияти ин усул аз он иборат аст, ки як массиви нав муаррифӣ карда шуда, дар он ба сифати элемент массиви аввала қарор дода мешаванд. Сипас, дар ҳар қучое, ки хочати ду элементро ба ҳамдигар муқоиса намудан пеш ояд, ин амал хело сода анҷом мегирад, зеро ба ин усул андозаи массив хурд карда мешавад. Мантиқи кори равиши мазкур ҳам монанди равиши хубобӣ мебошад. Мисол:

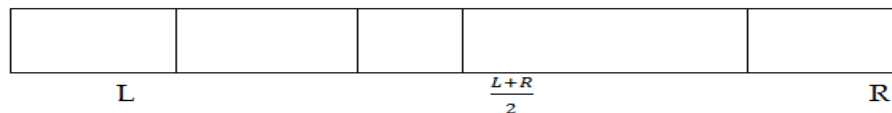
```
#include <iostream>
using namespace std;
int main()
{
    const int n=20;
    int i,j,k;
    int a[n],b[n],x;
    cout<<"Elementhoi massiv pesh az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++){
        a[i]=rand()%150;
        cout<<a[i]<<" ";
    }
    cout<<endl<<endl;
    for (k=0;k<n;++k)
        b[k]=k;
    for(i=1;i<n;i++)
        for(j=0;j<n-i;j++)
            if(a[b[j]]>a[b[j+1]])
            {
                x=b[j];
                b[j]=b[j+1];
                b[j+1]=x;
            }
    cout<<"Elementhoi massiv pas az
jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)
        cout<<a[b[i]]<<" ";
    cout<<endl;
    return 0;
}
```

Мураттабсозии босуръат ё зуд (quick sort)

Бигзор элементҳои массив ба сурати қадвале бузурге дода шуда бошанд. Барои онҳоро мураттаб намудан равиши нави бо номи мураттабсозии босуръатро дида мебароем. Моҳияти ин навъ мурат-табсозӣ аз он иборат аст, ки аввал як элементро аз қадвали додашуда интихоб намуда, элементҳои

боқимондае, ки аз он калонанд, дар тарафи рости он ва элементҳои боқимондаро бошад, дар тарафи чапи он қарор медиҳем. Дар натиҷа ҷадвали мо ба ду ҷадвал тақсим карда мешавад. Амали мазкурро бо он ҷадвалҳо низ иҷро мекунем, ки дар натиҷа ҷадвалҳои нав ҳосил карда мешаванд. Ин амалро то замоне иҷро мекунем, ки элементҳои ҷадвал ба таври зиёдшавиашон ҷо ба ҷо гузошта шаванд.

Элементе, ки дар ибтидо мо интиҳоб кардем, шояд баъзе хонандагони арҷманди мо дар бораи чи тавр интиҳоб кардани он суол кунанд. Дар аввал ҷадвалро ба намуди уфуқӣ тасаввур карда, хонаи аввал ё самти чапи онро бо L ва хонаи охир ё самти рости онро бо R ишора мекунем, яъне



Ду тағйирёбандаи i ва j интиҳоб намуда, тағйирёбандаи i -ро ба L ва тағйирёбандаи j -ро ба R баробар қарор дода, элементҳои миёнаро ҷунин интиҳоб мекунем.

$$x = a \left[\frac{L + R}{2} \right]$$

Баъд аз он то замоне (мавқеъе), ки $a[i] < x$ ва $i < R$ будан, тағйирёбандаи i -ро ба тарафи рост ҳаракат медиҳем. Ин амалро барои тағйирёбандаи j низ иҷро мекунем, яъне то замони $a[j] > x$ будан тағйирёбандаи j -ро ба тарафи чап ҳаракат медиҳем, баъд аз он, ҷойи $a[i]$ -ро ба $a[j]$ иваз мекунем, сипас ба тағйирёбандаи i як воҳид илова намуда, аз тағйирёбандаи j як воҳид кам мекунем ва то замони $i \leq j$ -шудан ин амал такрор карда мешавад. Элементе, ки ба сифати элементҳои мобайнӣ интиҳоб намуда, будем онро дар дохили ҷадвал ҳаракат дода, дар охир дар ҷойи худ қарор медиҳем. Ба ҳамин тариқ, элементҳои аз он хурдбуда дар тарафи чапи он ва элементҳои аз он калон буда дар тарафи росташ қарор мегиранд. Дар натиҷа, элементҳои массив мураттаб карда мешаванд.

Мисол. Барномае нависед, ки элементҳои массиви додашударо ба равиши мураттабсозии босуръат (quick sort) мураттаб намуда, дар функцияи асли ба ҷоп диҳад.

```
#include <iostream>
using namespace std;
void QuickSort(int a[],int avval,int ohir);
int main()
{
    const int n=25;
    int i,j,avval=0,ohir=n-1,x,k,t;
    int a[n],b[n];
    cout<<"Elementhoi massiv pesh az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)
    {
        a[i]=rand()%105;
        cout<<a[i]<<" ";
    }
    cout<<endl<<endl;
```



```

QuickSort(a,0,n-1);
cout<<"Elementhoi massiv pas az jobajoguzori"<<endl<<endl ;
for (i=0;i<n;i++)
    cout<<a[i]<<" ";
cout<<endl;

return 0;
}
void QuickSort(int a[],int avval,int ohir)
{
    int i=avval;
    int j=ohir;
    int t, x;
    x=a[(avval+ohir)/2];
    while (i<=j)
    {
        while(a[i]<x && i<ohir)
            i++;
        while( a[j]>x && j>avval)
            j--;
        if(i<=j)
        {
            t=a[i];
            a[i]=a[j];
            a[j]=t;
            i++;
            j--;
        }
        if (j>avval)
            QuickSort(a,avval,j);
        if(i<ohir)
            QuickSort(a,i,ohir);
    }
}

```

Чустучӯ кардан(searching)

Баъзан лозим меояд, ки дар дохили маҷмӯе аз элементҳо элементро чустучӯ кунем. Масалан, чустучӯи калима дар фарҳанги луғат. Барои онро ба даст овардан ҷойи лозимии фарҳангро чустучӯ мекунем. Барои ба таври электронӣ чустучӯ кардан якчанд алгоритм ё роҳҳои мавҷуд аст, ки онҳо ба мо имконият медиҳанд, ки аз дохили маҷмӯи элементҳои лозимиро дарёфт намоем. Алгоритмҳои асосии чустучӯи инҳо мебошанд:

- 1) Чустучӯи хатти(liner search)
- 2) Чустучӯи равиши дӯӣ (binary search)
- 3) Тасодуфӣ

Чустучӯи хаттӣ

Яке аз роҳҳои содатарини чустучӯи элементе дар дохили массив ё чадвали номуруттаби додаҳо он аст, ки элементи мавриди назарро ба элементҳои яқум, дуҷум ва ҳоказо n -ум муқоиса мекунем, агар ба ягон элементи маҷмӯъ баробар бошад, он гоҳ шумораи онро ба хотир гирифта, ба амали чустучӯи хотима мебахшад. Ва дар ҳолати акс пайғоми муносибе ҷоп мегардад.

```
#include <iostream>
using namespace std;
int Hatti_search(int a[],int n,int x)
{
    int i;
    for (i=0;i<n;i++)
        if (x==a[i])
            return a[i];
    return 0;
}
int main()
{
    const int n=25;
    int i,search_key;
    cout<<"search_key=";<<cin>>search_key;
    int a[n];
    cout<<"\nElementhoi massiv\n\n" ;
    for (i=0;i<n;i++){
        a[i]=rand()%105;
        cout<<a[i]<<" ";
    }
    cout<<endl<<endl;
    if (Hatti_search(a,n,search_key))
        cout<<"Elementi mavridi
nazar="<<(Hatti_search(a,n,search_key));
    else
        cout<<"Chunin element vujud nadorad!";
    cout<<endl;
    return 0;
}
```

Чустучӯи равиши дӯй (binary search)

Яке аз равишҳои содаи дигари чустучӯи элементҳои массив дар дохили чадвал равиши дӯй мебошад. Равиши мазкур дар мавриде, ки элементҳои массив муруттабанд, ба қор бурда мешавад. Дар муқоиса нисбат ба равиши хаттӣ сарътар мебошад.

Алгоритми ин равиш чунин аст: дар аввал ибтидо ва интиҳои чадвалро ба тағйирёбандаҳои l ва m нисбат дода, элементи мавриди назарро бо элементи $m = \frac{l+h}{2}$ муқоиса мекунем, агар баробар бошад, пас, он пайдо шудааст ва дар акси ҳол, агар элементи мавриди назар аз элементи m -ум калонтар бошад, тағйирёбандаи h -ро ба $m+1$ баробар қабул карда мешавад. Ва дар ҳолати

элементи мазкур аз элементи m -ум хурд будан, h –ро ба $m-1$ қарор медиҳем ва элементи мавриди назарро боз бо элементи мобайнии нав ёфтамон муқоиса мекунем. Ин амалро то замоне идома медиҳем, ки шарти $L \leq h$ дуруст бошад. Баъд амали мазкур хотима меёбад ва элементи мавриди назар ба ҷоп дода мешавад. Мисол: функсияе нависед, ки аз элементҳои массиви бо ягон равиш мураттабшуда, элементи лозимиро бо равиши бинари ҷустуҷӯ намуда барорад.

```

#include <iostream>
using namespace std;
int Binary_search(int a[],int n,int x){
    int m,l=0,h=n-1;
    while (l<=h)
    {
        m=(l+h)/2;
        if (x==a[m])
            return a[m];
        if (x>a[m])
            l=m+1;
        else
            h=m-1;
    }
    return 0;
}
int main(){
    const int n=25;
    int a[n];
    int i,b,j,search_key;
    cout<<" Elementhoi massiv pesh az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)
    {
        a[i]=rand()%45;
        cout<<a[i]<<" ";
    }
    cout<<endl;
    for (j=1;j<n;++j)
        for (i=0;i<n-j;++i)
            if (a[i]>a[i+1])
            {
                b=a[i];
                a[i]=a[i+1];
                a[i+1]=b;
            }
        cout<<endl;
    cout<<"Elementhoi massiv pas az jobajoguzori"<<endl<<endl ;
    for (i=0;i<n;i++)
        cout<<a[i]<<" ";
    int d=0;
    while(++d<3)
    {

```

```

        cout<<"\n\nsearch_key=";<cin>>search_key;
        if ( Binary_search(a,n,search_key))
cout<<"\nElementi mavridi nazar="<<Binary_search(a,n,search_key );
        else
        cout<<"\nChunin element vujud nadorad!\n";
    }
    return 0;
}

```

Тавре ки дида мешавад, дар ин ҷо аввал элементҳои массив бо равиши хубобӣ мураттаб карда шуда, баъд бо ёрии фурӯхонии функсияи Binary_search() элементҳои лозимиеро аз маҷмӯа ҷустуҷӯ намуда, баъди ёфтани он ба ҷоп дода шудааст.

Ҷустуҷӯи равиши тасодуфӣ

Моҳияти равиши мазкур аз он иборат аст, ки массиви додашуда ба се қисмат ҷудо карда мешавад. Элементҳои аз адади тасодуфии интихобкарда калон, элементҳои рақамашон аз адади тасодуфии хурд ва элементҳои ба адади тасодуфии рақами тартибиашон баробар. Адади тасодуфӣ дар фосилаи аз 1 то n, ки дар ин ҷо n андозаи массив мебошад, интихоб карда мешавад. Баъд вобаста аз миқдори элементҳо дар ҳар як қисмат барои ҷустуҷӯ як қисмат интихоб карда мешавад.

Массивҳои бисёрченака

Тавре ки қаблан гуфтем, массивҳо якченака дученака ва n-ченака мешаванд. Массивҳои бисёрченака ҳам монанди массивҳои якченака муаррифӣ карда мешаванд, вале фарқи онҳо дар он аст, ки бузургии (ҳар як ченак) ченакҳои массив дар дохили n- қавси фигу-рави навишта мешавад. Наҳви муаррифӣ намудани чунин массивҳо шакли зеринро дорад:

data_type array_Name [size1][size2]...[size n]

Дар ин ҷо data_type навъи массив, array_Name-номи массив ва size1, size2, ..., size n мувофиқан андоза (ченак)-ҳои яқум, дуюм ва ҳоказо n-ум мебошанд. Барои ба массивҳои n-ченака шинос шудан ҳолати n=2 буданро дида мебароем. Бигзор ба мо ҷадвале дорои n- сатр ва m –сутун дода шуда бошад. Дар ин ҷо n-миқдори сатрҳо ва m-миқдори сутунҳо мебошад. Барои ҳар як элементҳои ин ҷадвал дастёб шудан бояд, ки суроғи ҳар як катакчаро донем. Аз нуқтаи назари математика массивҳои дученака ин матритсаи дорои n сатр ва m сутун мебошанд. Ҳангоми n=3 ва m=3 будан, матритсаи шакли зеринро мегирад:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Тавре ки дида мешавад, дар ин ҷо ҳар як элементҳои матритсаи дорои ду зернавис буда, якеаш нишондиҳандаи сатр ва дигараш нишондиҳандаи сутун мебошад. Масалан, барои элементҳои сеюми сатри дуввумро ба тағйирёбандаи t бахшидан чунин менависем: t=a₂₃ Дар ин ҷо адади 2 нишондиҳандаи сатр ва 3 бошад, нишондиҳандаи сутун мебошад.

Наҳви муаррифӣ намудани массиви дученака шакли зеринро дорад:

data_type array_Name [n][m]

Фарз мекунем, ки $n=2$ ва $m=3$ мебошад, он гоҳ мастритсаи мазкур чадвалеро мемунад, ки дорои ду сатр ва се сутун мебошад, яъне

a[0][0]	a[0][1]	a[0][2]	a[0][3]
a[1][0]	a[1][1]	a[1][2]	a[1][3]
a[2][0]	a[2][1]	a[2][2]	a[2][3]



Дар ҳолати умумӣ массиви дорои n -сатр ва m сутунро массиви n бар m ($n - in - m array$) мегӯянд. Массивҳои сеченака чорченака ва ғай-ра низ ҳамин тавр муаррифӣ карда мешаванд. Корбурди ин гуна мас-сивҳо барои истифодабарандаи оддӣ камтар аст, аз ин рӯ, онҳоро сарфи назар карда, фақат массивҳои дученакаро дида мебароему бас.

Ба массивҳои дученака бахшидани қимати аввала

Монанди массивҳои якченака ба массивҳои дученака низ ҳан-гоми муаррификунӣ метавон қимати аввала бахшид. **Мисол.**

```
int array[2][3]={1,2,3,4,5,6}
```

Агар ин массивро ба намуди чадвал тасвир кунем намуди зеринро хоҳад гирифт:

1	2	3
4	5	6

дар ин ҷо:

```
array[0][0]=1, array [1][0]=2, array [0][2]=3  
array [1][0]=4, array[1][1]=5, array [1][2]=6
```

Бояд дар ёд дошт, ки зернависи элементҳои массив ду ченака низ аз 0 сар мешавад. Ба ғайр аз ин навъ муаррификунӣ, элементҳои ҳар як сатр метавонанд дар дохили як қавси фигуравӣ навишта шаванд:

```
int array[2][3]={  
  {1,2,3},  
  {4,5,6}  
};
```

Агар дар дохили ҳар қавси фигуравӣ элементҳои додашуда аз андозаи массив кам бошанд, элементҳои боқимонда ба таври пешфарз қимати 0 қабул мекунанд. **Мисол:**

```
int array[2][3]={  
  {1},  
  {4}  
};
```

Дар ин чо:

array[0][0]=1, array [1][0]=0, array [0][2]=0

array [1][0]=4, array[1][1]=0, array [1][2]=0

Истифодаи массивҳои дученака

Аз нуқтаи назари математика, Тавре ки қаблан гуфтем, массивҳои дученака ин матритсаҳо мебошанд. Матритса ин ҷадвали дорои n -сатр ва m -сутун мебошад. Одатан рақами сатр бо i ва рақами сутун бо j ишора карда мешавад. Тавре ки қаблан зикр гардид, зернавис (рақами тартиби)-и 0 мебошад. Пас, аз ин чо ин матритса ҷадвали до-рои $n-1$ сатр ва $m-1$ -сутун мебошад. Мисол `int matr[10][10]` матритсае аст, ки аз 10 сатр ва 10 сутун иборат мебошад.

Монанди массивҳои якченака массивҳои дученака низ ба барнома (компютер) дохил ва хориҷ карда мешавад.

Мисол. Барномае нависед, ки элементҳои матритсаро дохил намуда, Сипас, ба сурати ҷадвал ба ҷоп диҳад.

```
#include <iostream>
using namespace std;
int main()
{
    const int n=4,m=3;
    int matr[n][m];
    int i,j;
    for (i=0;i<n;i++)
        for(j=0;j<m;j++)
            cin>>matr[i][j];
    for (i=0;i<n;i++)
    {
        for (j=0;j<m;j++)
            cout<<matr[i][j]<<"\t";
        cout<<endl;
    }
    return 0;
}
```

Натиҷаи барнома ба сурати зерин хоҳад буд.

1	0	4	7	5	3	9	21	4	5	22	58
Matrix											
1						0					4
7						5					3
9						21					4
5						22					58

Матритсаҳои рамзӣ

Монанди массивҳои якченака, элементҳои массиви дученака метавонанд низ рамзҳо бошанд. Дар ин навъ массивҳо зернависи яқум нишондиҳандаи сатр буда, зернависи дуввум дарозии сатрро нишон медиҳад.

Мисол. Барномае нависед, ки номи рангҳои додашударо барорад.

```

#include <iostream>
using namespace std;
int main()
{
    const int n=3,m=5;
    int i,j;
    char Color[n][m]={
        {'R','E','D'},
        {'G','R','E','E','N'},
        {'B','L','U','E'},
        {'B','L','A','C','K'},
    };
    for (i=0;i<n;i++)
    {
        for (j=0;j<m;j++)
            cout<<Color[i][j];
        cout<<endl;
    }
    return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

RED
GREEN
BLACK

Ғалбери Эратосфен

Барои муайян кардани ададҳои сода алгоритми махсусе мавҷуд аст, ки онро ғалбери Эратосфен (Эратосфён Кирёнский (Ἐρατοσθένης ὁ Κυρηναῖος; 276 п.м.—194 п.м.) —математик, астроном, географ ва шоири Юнон) меноманд. Моҳияти ин алгоритм чунин аст:

- 1) Ададҳои аз як то n -ро пайҳам менависем;
 - 2) Адади якро хат мезанем;
 - 3) Адади 2-ро интиҳоб карда (дар дохили давра гирифта), ададҳои бо он каратиро хат мезанем;
 - 4) Барои адади 3 низ қадами сеюмро иҷро мекунем;
 - 5) Барои ададҳои боқимонда, ки каратии онҳо хат зада нашудаанд, қадами сеюмро иҷро мекунем;
 - 6) Пас аз хат задани ададҳо, танҳо ададҳои сода боқӣ мемонанд.
- Мисоли ин алгоритм дар поён оварда шудааст.

```

#include <iostream>
using namespace std;
const int N=1000;
unsigned long long int a[N];
int main(){
    long int i;
    for(long int i=0; i<N; i++)
        a[i] = i;
    a[1]=0;
}

```

```

for(long int s=2; s<N; s++)
    if(a[s]!=0)
        for(long int j=s*2; j<N; j+=s)
            a[j]=0;
for(i=0; i<N; i++)
    if(a[i]!=0)
        cout<<a[i]<< " ";
return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89
97 101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179
181 191 193 197 199 211 223 227 229 233 239 241 251 257 263 269 271
277 281 283 293 307 311 313 317 331 337 347 349 353 359 367 373 379
383 389 397 401 409 419 421 431 433 439 443 449 457 461 463 467 479
487 491 499 503 509 521 523 541 547 557 563 569 571 577 587 593 599
601 607 613 617 619 631 641 643 647 653 659 661 673 677 683 691 701
709 719 727 733 739 743 751 757 761 769 773 787 797 809 811 821 823
827 829 839 853 857 859 863 877 881 883 887 907 911 919 929 937 941
947 953 967 971 977 983 991 997

```

Ғалбери Сундарама

Барои ёфтани ададҳои содда якчанд алгоритмҳо вучуд дорад. Яке аз чунин алгоритмҳои хеле машҳур ғалбери Эратосфен мебошад. Новабаста аз маъмул будани алгоритми мазкур суръати муайянкунии ададҳои содда дар он суст мебошад. Соли 1934 алгоритми наво номи ғалбери Сундарама пайдо шуд, ки нисбат ба ғалбери Эратосфен тезтар иҷро карда мешавад. Хосияти асосии алгоритми мазкур аз он иборат аст, ки бо дода шудани адади n ададҳои соддаи фосилаи 1 то $2 \cdot n$ муайян карда мешаванд.

Дар ин алгоритм аз ду ҳалқаи такрорӣ истифода бурда мешавад, ки ҳалқаи такрорӣ берун то замоне, ки қимати $2 \cdot i \cdot j + i + j$ хурд аз n будан иҷро карда мешавад. Ҳалқаи такрорӣ дохилӣ бошад, то замоне ки шарт $j \leq (n - i) / (2 \cdot i + 1)$ дорой қимати ҳақ аст, иҷро карда мешавад. Барои ба моҳияти алгоритми мазкур шинос шудан барномаи ин алгоритмро менависем.

```

#include <iostream>
using namespace std;
const int n=500;
int x[2*n];
int main()
{
    int i, j;
    cout<<"Adadhoi sodda:"<<endl<<2<<" ";
    for (i=1; i<=n; i++) x[i]=1;
    i=1; j=1;
    while ((2*i*j+i+j)<=n)
    {
        while (j<=(n-i)/(2*i+1))

```



```

        {
            x[2*i*j+i+j]=0;
            j++;
        }
        i++; j=i;
    }
    for (i=1; i<=n; i++)
        if (x[i])
            cout<<2*i+1<<" ";
    return 0;
}

```

Намунаи ҳалли масъалаҳо

1. Массиви a , ки дорои n элемент мебошад, дода шудааст. Барномае нависед, ки миқдор ва суммаи элементҳои ҷуфти онро ҳисоб кунад.

```

#include <iostream>
using namespace std;
int main(){
    const int n=10;
    int a[n];
    int i,k=0;
    long int S=0;
    cout<<" Masiivi dodoshuda\n";
    for (i=0;i<n;i++)
        cin>>a[i];
    for (i=0;i<n;i++)
        if(a[i]%2==0)
        {
            k++ ;
            S+=a[i];
        }
    cout<<"S="<<S<<"\nk="<<k<<endl;
    return 0;
}

```

2. Адади натуралии N дода шудааст. Чунин массиве тартиб диҳед, ки элементҳои аз ададҳои токи фосолаи аз як то N иборат бошанд.

```

#include <iostream>
using namespace std;
int main(){
    const int n=20;
    int a[n]={0};
    int i,nol,kol=0;
    long int S=0;
    cout<<" Masiivi dodoshuda\n";
    for (i=0;i<n;i++)
        if (i%2!=0)
            a[i]=i;
    for (i=0;i<n;i++)
        if (a[i]!=0)

```

```

        cout<<a[i]<<"\t";
    return 0;
}

```

3. Адади натуралии N ($N > 2$) дода шудааст. Массиве тартиб диҳед, ки элементҳои он аз N ададҳои аввалии Фибоначӣ иборат бошанд. Маълум аст, ки ададҳои Фибоначӣ намуди зеринро доранд.

$$F_1 = 1, F_2 = 1, F_K = F_{K-2} + F_{K-1}$$

```

#include <iostream>
using namespace std;
int main()
{
    const int n=10;
    int a[n]={0};
    int i,nol,kol=0;
    long int S=0;
    cout<<" Adadhoi Fibonacci\n";
    a[0]=1;a[1]=1;
    for (i=2;i<n;i++)
        a[i]=a[i-2]+a[i-1];
    for (i=0;i<n;i++)
        cout<<a[i]<<"\t";
    return 0;
}

```

Массиви якченакаи андозааш N ва ададҳои бутуни K ва L ($1 < K \leq L \leq N$) дода шудааст. Суммаи ҳамаи элементҳои аз K то L –и массив ҳисоб карда шаванд.

```

#include <iostream>
using namespace std;
int main(){
    const int n=20;
    int a[n]={0};
    int i,L,K;
    long int S=0;
    cout<<"K=";cin>>K;
    cout<<"L=";cin>>L;
    for(i=0;i<n;i++){
        a[i]=rand()%25;
        cout<<a[i]<<"\t";
    }
    for (i=K;i<=L;i++)
        S+=a[i];
    cout<<"S="<<S<<"\t";
    return 0;
}

```

4. Массиви якченакаи андозааш N дода шудааст. Дар аввал элементҳои манфӣ, баъд мусбат ва дар охир элементҳои нулии онро бароред.

```

#include <iostream>
using namespace std;
int main(){
    const int n=20;
    int a[n],Z[n];
    int i,k=-1;
    long int S=0;
    cout<<"Massivi dodashuda\n\n";
    for(i=0;i<n;i++)
    {
        a[i]=rand()%10-5;
        cout<<a[i]<<"\t";
    }
    cout<<endl;
    cout<<"Massivi nav\n\n";
    for (i=0;i<n;i++)
        if(a[i]<0)
        {
            ++k;
            Z[k]=a[i];
        }
    for (i=0;i<n;i++)
        if(a[i]>0)
        {
            k++;
            Z[k]=a[i];
        }
    for (i=0;i<n;i++)
        if(a[i]==0)
        {
            k++;
            Z[k]=a[i]; }
    for (i=0;i<n;i++)
        cout<<Z[i]<<"\t";

    return 0;
}

```

5. Массиви якченакаи андозааш n дода шудааст. Барномае нависед, ки элементҳои массивро чунин иваз кунад: элементҳои якумро ба элементҳои n -ум, элементҳои дуҷумро ба элементҳои $n-1$ -ум ва ҳоказо.

```

#include <iostream>
using namespace std;
int main(){
    const int n=10;
    int i,j;
    int A[n],b;
    for (i=0;i<n;i++)
    {
        A[i]=rand()%(25);
        cout<<"A["<<i<<"]="<<A[i]<<" \n ";
    }
}

```

```

    }
    cout<<endl;
    for (i=0;i<=n/2;b=A[i],A[i]=A[n-1-i],A[n-1-i]=b,i++);
    for (i=0;i<n;i++)
        cout<<"A["<<i<<"]="<<A[i]<<" \n ";
    return 0;
}

```

6. Массиви якченакаи тартибаш n дода шудааст. Барномае нависед, ки m элементи калонтарини массивро бароред.

```

#include <iostream>
using namespace std;
int main(){
    const int n=20,m=5;
    int a[n],b[m];
    int i,j,minmax;
    for (i=0;i<n;i++)
        a[i]=rand()%80;
    for (i=0;i<n;i++)
        cout<<a[i]<<"\t";
    cout<<endl;
    for(i=0;i<m;i++)
        b[i]=a[i];
    for(i=m+1;i<n;i++)
    {
        minmax=0;
        for (j=1;j<m;j++)
            if (b[j]<b[minmax])
                minmax=j;
        if (a[i]>b[minmax])
            b[minmax]=a[i];
    }
    for (i=0;i<m;i++)
        cout<<b[i]<<"\t";
    return 0;
}

```

7. Барномае нависед, ки адади бутуни мусбати m –ро аз намуди даҳйба намуди дӯй гардонад.

```

#include <iostream>
using namespace std;
int main(){
    int a[10]={0},n,i=0;
    cout<<"n=";cin>>n;
    while(n>=1)
    {
        a[i]=n%2;
        n/=2;
        ++i;
    }
    for(;i>=0;--i)

```

```

        cout<<a[i];
    return 0;
}

```

8. Массиви якченакаи тартибаш n дода шудааст. Барномае нависед, ки m элементи хурдтарини массивро бо тартиби зиёдшавиаш барорад.

```

#include <iostream>
using namespace std;
int main(){
    const int n=50,m=15;
    int a[n],b[m],temp;
    int i,j,minmax;
    for(i=0;i<n;i++)
    {
        a[i]=rand()%80;
        cout<<a[i]<<"\t";
    }
    cout<<endl;
    for(i=0;i<m;i++)
        b[i]=a[i];
    for (i=m;i<n;i++)
    {
        minmax=0;
        for(j=1;j<m;j++)
            if(b[j]>b[minmax] )
                minmax=j;
        if(a[i]<b[minmax])
            b[minmax]=a[i];
    }
    for(j=1;j<m;j++)
        for(i=0;i<m-j;i++)
            if(b[i]>b[i+1])
            {
                temp=b[i];
                b[i]=b[i+1];
                b[i+1]=temp;
            }
        for (i=0;i<m;i++)
            cout<<b[i]<<"\t";
        cout<<endl;
    return 0;
}

```

9. Матритсаи дорои n сатр ва m сутун дода шудааст. Барномае нависед, ки элементҳои диагонгали асосии онро ба элементи диагонали баракс (ёрирасон) иваз кунед.

```

#include <iostream>
using namespace std;
int main(){
    const int n=3,m=3;
    int a[n][m],i,j,k,temp;

```

```

double S;
for(i=0;i<n;i++)
    for(j=0;j<m;j++)
        a[i][j]=rand()%50;
for(i=0;i<n;i++)
{
    for(j=0;j<m;j++)
        cout<<a[i][j]<<"\t";
    cout<<endl;
}
cout<<endl;
for(i=0;i<n;i++)
{
    temp=a[i][i];
    a[i][i]=a[i][n-1-i];
    a[i][n-1-i]=temp;
}
for(i=0;i<n;i++)
{
    for(j=0;j<m;j++)
        cout<<a[i][j]<<"\t";
    cout<<endl;
}
return 0;
}

```

10. Матритсаи дорои n сатр ва m сутун дода шудааст. Барномае нависед, ки элементҳои ҳар як сатри онро ба тартиби калонша-виашон барорад.

```

#include <iostream>
using namespace std;
int main(){
    const int n=3,m=3;
    int a[n][m],i,j,k,temp;
    double S;
    for(i=0;i<n;i++)
        for(j=0;j<m;j++)
            a[i][j]=rand()%50;
    for(i=0;i<n;i++){
        for(j=0;j<m;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    cout<<endl;
    for(i=0;i<n;i++)
        if(j%2==0)
        {
            for(k=1;k<m;k++)
                for (j=0;j<m-k;j++)
                    if(a[i][j]>a[i][j+1])
                        temp=a[i][j];

```

```

        a[i][j]=a[i][j+1];
        a[i][j+1]=temp;
    }
    else
        for(k=1;k<m;k++)
            for (j=0;j<m-k;j++)
                if (a[i][j]<a[i][j+1])
                {
                    temp=a[i][j];
                    a[i][j]=a[i][j+1];
                    a[i][j+1]=temp;
                }
            for(i=0;i<n;i++)
            {
                for(j=0;j<m;j++)
                    cout<<a[i][j]<<"\t";
                cout<<endl;
            }

    return 0;
}

```

11. Матритсаи дорои n сатр ва m сутун дода шудааст. Элементҳои аввалии ҳар як сатро ба қимати миёнаи арифметикии ҳар сатр иваз кунед.

```

#include <iostream>
using namespace std;
int main(){
    const int n=3,m=15;
    int a[n][n],i,j;
    double S;
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            a[i][j]=rand()%50;
    for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    cout<<endl;
    for(i=0;i<n;a[i][0]=S/n,i++)
        for(S=j=0;j<n;j++)
            S+=a[i][j];
    for(i=0;i<n;cout<<endl,i++)
    {
        for(j=0;j<n;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    return 0;
}

```

12. Матритсаи дорои n сатр ва m сутун дода шудааст. Муайян кунед, ки оё он воҳиди аст?

Қайд. Матритса воҳидӣ номида мешавад, агар дар диагонали асосии он воҳидҳо ва боқимонда элементҳои он аз нолҳо иборат бошанд.

```
#include <iostream>
using namespace std;
int main(){
    const int n=3,m=15;
    int a[n][n],i,j,pr;
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            a[i][j]=rand()%50;
    for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    cout<<endl;
    for(pr=1,i=0;i<n;i++)
        for(j=i;j<n;j++)
            if(((i==j) && (a[i][j]!=1)) || ((i!=j) && (a[i][j]!=0)))
            {
                pr=0;
                break;
            }
        if (pr)
            cout<<"Vohidi ast \t";
        else
            cout<<"Vohidi nest \t";
        cout<<endl;
    }
    return 0;
}
```

13. Матритсаи дорои n сатр ва m сутун дода шудааст. Суммаи элементҳои аз диагонали асосӣ поёнро ҳисоб кунед.

```
#include <iostream>
using namespace std;
int main(){
    const int n=3 ;
    int i,j;
    int a[n][n];
    for (int i=0;i<n;i++)
        for(int j=0;j<n;j++)
            cin>>a[i][j];
    for (int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
}
```



```

    }
    for (int i=0;i<n;i++)
    {
        for (j=0;j<i;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    return 0;
}

```

14. Массиви индозааш N дода шудааст. Барномае нависед, ки элементҳои диагонали баракси массивро барорад.

```

#include <iostream>
using namespace std;
int main(){
    const int n=3 ;
    int i,j;
    int a[n][n];
    for (int i=0;i<n;i++)
        for(int j=0;j<n;j++)
            cin>>a[i][j];
    for (int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    cout<<" \n ";
    for (int i=0;i<n;i++){
        for (j=n-i;j<n;j++)
            cout<<a[i][j]<<"\t";
        cout<<endl;
    }
    return 0;
}

```

15. Ададҳои бутуни аз 1 то n^2 -ро дар матритсаи дученакаи квадратии андозааш ба n баробар ба таври морпеч нависед. Намуна: ҳангоми $n=5$.

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

Матни барнома.

```

#include <iostream>
using namespace std;
int main(){
    int const n=5;
    int A[n][n];

```

```

int i=1,j,k,p=n/2;
for(k=1;k<=p;k++)/*давр бо рақами гардиш (витка)*/
{
/*Муайянкунии қимати болоии қуллаи сутунҳо*/
for (j=k-1;j<n-k+1;j++) A[k-1][j]=i++;
/* Тарафи рости амудии сутунҳо*/
for (j=k;j<n-k+1;j++) A[j][n-k]=i++;
/* Тарафи поёнии уфуқии сутунҳо */
for (j=n-k-1;j>=k-1;--j) A[n-k][j]=i++;
/*Тарафи чапи амудии сутунҳо*/
for (j=n-k-1;j>=k;j--)
A[j][k-1]=i++;
}
if (n%2==1)
A[p][p]=n*n;
for(i=0;i<n;i++)
for(j=0;j<n;j++)
{
cout<<A[i][j]<<" ";
if(j==n-1)
cout<<endl;
}
return 0;
}

```

Масъалаҳо барои кори мустақилона **Масъалаҳои гурӯҳи Аггау**

1. Массиви a , ки элементҳояш ададҳои ҳақиқӣ мебошанд, дода шудааст. Барномае нависед, ки суммаи қисми касрии элементҳои онро ҳисоб кунад.
2. Массиви a , ки дорои 50 элемент мебошад, дода шудааст. Барномае тартиб диҳед, ки аввал элементҳои ба ду қаратӣ ва баъд элементҳои ба се қаратии онро хориҷ кунад.
3. Массиви a , ки аз ададҳои ҳақиқӣ иборат мебошад, дода шудааст. Барномае тартиб диҳед, ки сумма ва ҳосили зарбии қисмҳои бутуни массиви додасударо ҳисоб кунад.
4. Массиви $a(15)$ дода шудааст. Сумма ва миқдори элементҳое, ки дар байни элементҳои минималӣ ва максималиро меҳобанд, ёфта шавад.
5. Массиви $A(12)$ дода шудааст. Элементҳои токи онро хат занед ва квадрати элементҳои боқимондари ҳисоб кунед.
6. Массиви a , ки дорои n элемент мебошад, дода шудааст. Барномае нависед, ки миқдор ва суммаи элементҳои чуфти онро ҳисоб кунад.
7. Адади натуралии N ($N > 2$) дода шудааст. Массиве тартиб диҳед, ки элементҳои он аз N ададҳои аввалии Фибоначӣ иборат бошанд. Маълум аст, ки ададҳои Фибоначӣ шакли зеринро доранд.

$$F_1 = 1, F_2 = 1, F_k = F_{k-2} + F_{k-1} \quad (k > 3)$$

8. Массиви якченакаи андозааш N ва ададҳои бутуни K ва L ($1 < K \leq L \leq N$) дода шудааст. Суммаи ҳамаи элементҳои аз K то L –и массив ҳисоб карда шаванд.

9. Массиви якченакаи андозааш N дода шудааст. Дар аввал элементҳои манфӣ, баъд мусбат ва дар охир элементҳои нулионро бароред.

10. Массиви ададҳои ҳақиқӣ дода шудааст. Ба ҳамаи элементҳои рақами тартибиашон тоқ адади 1 чамъ карда шавад, аз ҳамаи элементҳои рақами тартибиашон чуфт бошад адади 1 тарҳ карда шавад.

11. Массиви ададҳои ҳақиқӣ дода шудааст. Ба ҳамаи элементҳои манфии массив қимати элементи якум чамъ карда шавад, аз ҳамаи элементҳои қиматашон нол адади b тарҳ карда шавад ва элементҳои мусбати он бетағир гузошта шаванд (a, b, c -ададҳои ихтиёрӣ).

12. Массиви ададҳои ҳақиқӣ дода шудааст. Аз ҳамаи элементҳои мусбати массив адади a тарҳ карда шавад, аз ҳамаи элементҳои манфии массив адади b тарҳ карда шавад ва ба ҳамаи элементҳои қиматашон нол адади c чамъ карда шавад (a, b, c -ададҳои ихтиёрӣ).

13. Массиви якченакаи андозааш n дода шудааст. Барномае нависед, ки элементҳои массивро чунин иваз кунад: элементи якумро ба элементи n -ум, элементи дуумро ба элементи $n-1$ -ум ва ҳоказо.

14. Массиви якченакаи тартибаш n дода шудааст. Барномае нависед, ки m элементи калонтарини массивро бароред.

15. Муайян карда шавад, ки массив ба тарзи афзуншавӣ тартиб дода шудааст ё не. Агар не, пас элементе, ки ин қонуниятро (тартибро) вайрон мекунад хориҷ карда шавад.

16. Массиви ададҳои бутун дода шудааст. Ҳамаи элементҳое, ки қиматашон чуфт ва мавқеи ҷойгиршавиашон низ чуфт аст партофта карда шаванд.

17. Барномае нависед, ки адади бутуни мусбати m –ро аз намуди даҳӣба намуди дӯӣ гардонад.

18. Массиви якченакаи тартибаш n дода шудааст. Барномае нависед, ки m элементи хурдтарини массивро бо тартиби зиёдшавиаш барорад.

19. Адад дар системаи даҳӣ дода шудааст. Барномае нависед, ки онро ба системаи шонздаҳӣ гардонад.

20. Массив додашудааст. Ҳамаи элементҳои он ба элементи аввалаш (якумаш) зарб карда шаванд.

21. Усулҳои мурратабсозиро барои додаҳои тинҳои гуногун (сатрҳо, символҳо, ададҳои ҳақиқӣ ва ғайра) амалӣ соzed.

Масъалаҳои гурӯҳи Matrix

1. Массиви дученакаи $A[N][M]$ дода шудааст. Элементеро, ки дар сатр калонтарин ва дар сутун хурдтарин мебошад, ёфта шавад.

2. Суммаи элементҳои ҳар як сатри массиви дученакаи $A[N][M]$ –ро ҳисоб намуда дар массиви якченакаи $A[K]$ гирд оваред.

3. Суммаи элементҳои ҳар як сатри массиви дученакаи $A[N][M]$ –ро ҳисоб намуда, бо элементҳои диаганали асосӣ иваз кунед.

4. Массиви дученакаи $A[N][M]$ дода шудааст. Суммаи элементҳои ҳар як сутунро ҳисоб намуда, ба элементҳои диагонали баръакс иваз кунед.
5. Массиви дученакаи $A[N][M]$ дода шудааст. Элементҳои ҳар як сатри онро аз рӯи зиёдшавии қиматҳояшон ҷобачо гузоред.
6. Массиви дученакаи $A[N][M]$ дода шудааст. Элементҳои максималии ҳар як сатри онро ба элементи минималии ҳар як сутун иваз намоед.
7. Массиви дученакаи $A[N][M]$ ва $B[K][T]$ дода шудаанд. Элементҳои ин ду массивро дар массиви $C[X][Y]$ бо тартиби зиёдшавиашон гирд оваред.
8. Массиви дученакаи $A[N][M]$ дода шудааст, барномае нависед, ки элементҳои ҳар як сутуни онро мурағаб кунад.
9. Матритсаи дорои n сатр ва m сутун дода шудааст. Барномае нависед, ки элементҳои диагонали асосии онро ба элементи диагонали баръакс (ёрирасон) иваз кунед.
10. Матритсаи дорои n сатр ва m сутун дода шудааст. Барномае нависед, ки элементҳои ҳар як сатри онро ба тартиби калоншавиашон барорад.
11. Матритсаи дорои n сатр ва m сутун дода шудааст. Элементҳои аввалии ҳар як сатрро ба қимати миёнаи арифметикии ҳар сатр иваз кунед.
12. Матритсаи дорои n сатр ва m сутун дода шудааст. Суммаи элементҳои аз диагонали асосӣ поёнро ҳисоб кунед.
13. Матритсаи дорои n сатр ва m сутун дода шудааст. Муайян кунед, ки оё он воҳиди аст?
14. Массиви дученака дода шудааст. Ҳосили зарби ҳамаи элементҳои сутуни якуми массив ҳисоб карда шавад.
15. Массиви дученака дода шудааст. Суммаи квадратҳои ҳамаи элементҳои сутуни чоруми массив ҳисоб карда шавад.
16. Массиви дученака дода шудааст. Суммаи элементҳои тоқи массив ҳисоб карда шавад.
17. Массиви дученака дода шудааст. Миқдори элементҳои ҷуфт дар ҳар як сатри массив ёфта шавад.
18. Массиви дученака дода шудааст. Суммаи элементҳои ғайри мусбӣи ҳар як сатри массив ёфта шавад.
19. Массиви дученака дода шудааст. Рақами сутуне, ки дар он қимати хурдтарини массив ҷойгир аст ёфта шавад. Агар дар массив қимати хурдтарин якчандто мавҷуд бошад, пас ҳамоне, ки аз тарафи чап аввал омада аст ба назар гирифта шавад.
20. Массиви дученака дода шудааст. Рақами сатре ки дар он қимати калонтарини сутуни якуми массив ҷойгир аст ёфта шавад. Агар дар сутуни якум қимати калонтарин якчандто мавҷуд бошад, пас ҳамоне, ки аз боло аввал аст ба назар хориҷ карда шавад.
21. Массиви дученака дода шудааст. Координатаҳои элементи хурдтарини он ёфта шаванд.

Боби 7. Ишорагарҳо (Pointers)

Муқаддима

Яке аз хусусият ва имкониятҳои забони C/C++ истифода ва корбурди ишорагарҳо мебошад. Ишорагар ин тағйирёбандаест, ки ба ҷойи додае суроға ё маҳалли онро нишон медиҳад. Ба таври дигар гӯем, ишорагар тағйирёбандае мебошад, маҳал ва ё суроғаи додаи дигарро дар худ нигоҳ медорад, яъне миқдори ишорагар суроғае аз ҳофиза аст. Аз ҷо метавонем ба миқдори як дода ба таври ғайримустақим бо ёрии суроғааш аз ҳофиза дастёб шавем. Дар забонҳои кӯҳна ба монанди Algol 60, Gobol, Fortran ин навъ тағйирёбандаҳо пешбинӣ нашуда буд, аммо дар забонҳои Паскал, PL/1 ин гуна тағйирёбанда пешбинӣ шудааст. Забони C++/C забоне мебошад, ки нисбат ба забонҳои дигар аз ишорагарҳо ба таври васеътар истифода мебарад. Шояд хонандагони арҷманди мо суоле кунанд, ки истифодаи тағйирёбандаҳои навъи мазкур барои барномасоз чи имкониятеро фароҳам меоранд? Барои ба ин суол ҷавоб гуфтан матлабҳои зеринро дида мебароем:

1) Ҳангоми ба функсия интиқол намудани ягон дода, ба таври пешфарз нусхаи дода равон карда мешавад, ва ҳар амале, ки болои дода дар функсияи фаръӣ иҷро мегардад, ҳеҷ таъсире ба функсияи аслии намурузорад ва инчунин агар дода аз навъи массив ва ё сохтор буда, миқдори зиёд дошта бошад, ҳангоми нусхабардорӣ намудани он вақти зиёд сарф мешавад. Пас, барои содатар иҷро намудани ин амал аз тағйирёбандаҳои навъи ишорагар истифода мебарем. Тағйирёбандаи навъи ишорагар заминаи фуруҳонӣ ба суроға (call by address) ва ё фуруҳони бо нишона (call by reference)-ро фароҳам меорад, яъне он имконият медиҳад, ки ба функсияи фаръӣ на миқдори дода, балки суроғаи он интиқол дода шавад ва ҳар амале, ки болои додаи мазкур дар функсияи фаръӣ иҷро карда мешавад, таъсири зиёде болои он дода, дар функсияи аслии мегузорад.

2) Медонем, ки ҳар функсия танҳо як миқдорро бо номи худ бармегардонад. Агар лозим шавад, ки функсия якчанд миқдорро бар-гардонад, ба роҳи муқаррарӣ иҷрои ин амал ғайриимкон аст. Ишорагарҳо иҷрои ин амалро метавонанд ба роҳи содатар ва қулайтар иҷро кунанд, яъне ишорагарҳо суроғаи додаеро ба функсияи фаръӣ равон мекунанд ва миқдоре, ки функсияи фаръӣ бармегардонад дар суроғаи мазкур қарор мегирад.

3) Ишорагар бо массивҳо робитаи зиёд доранд. Дар натиҷа роҳи дигари дастёб шудан ба элементҳои массив фароҳам меорад. Инчунин ишорагар имконият медиҳад, ки як маҷмӯи сатрҳо дар дохили як массив танҳо намоиш дода шаванд, агарчанде, ки дарозии онҳо аз ҳамдигар фарқ кунанд низ.

4) Амали ҷудо намудани ҳофизаро ба таври айёни (dynamic storage allocation) имконпазир мегардонанд.

5) Тавассути ишорагарҳо сохтмони додаҳои динамикӣ, ки пушаҳо (стек), сафҳо (очеред), графҳо, дарахтҳои дӯи ва ғайра сохта мешаванд.

Метавон оиди ишорагарҳо мисолҳои зиёде орем, то хонандаи мо ба моҳияти тағйирёбандаҳои навъи мазкур пурратар шинос шавад. Масалан, мо ба одами наве шинос шуда, дар дафтarchai худ ном ва суроғаи онро сабт кардем.

Дар ин чо метавонем номи одамеро, ки бо он шинос шудем, як тағйирёбанда номида, суроғай онро барояш як ишорагар ҳисоб кард.

Бояд қайд кард, ки тавасути ишорагарҳо листҳои пайванди, графҳо, сафҳо, стекҳо ва ғайра сохта мешаванд.

Муаррифӣ намудани тағйирёбандаҳои навъи ишорагар

Монанди дигар тағйирёбандаҳо пеш аз истифода тағйирёбан-даҳои навъи ишорагар низ муаррифи карда мешаванд. Наҳви таърифи тағйирёбандаҳои ишорагар ба сурати зерин аст:

```
data_type *Ptvariable
```

Дар ин чо Ptvariable-номи тағйирёбанда буда, data_type-навъи тағйирёбанда аст, ки суроғай он метавонад дар тағйирёбандаи ишорагари Ptvariable қарор гирад. Рамзи ситорача («*») бошад, оператори ишорагар ба ҳисоб меравад, ки танҳо барои тағйирёбандаҳои ишорагар истифода мешавад. Ҳар тағйирёбандае, ки дар забони С++ истифода бурда мешавад, пеш аз истифода бояд муаррифӣ карда шавад ва дар вақти муаррифӣ як миқдор ҳофизаро банд мекунад, ки суроғай онро бо ёрии оператории ягонаи амперсанд (&) метавонем ба даст орем. Масалан, суроғай тағйирёбандаи n, ки қаблан аз ягон навъ муаррифӣ шудааст, бо ёрии ифодаи зерин ба даст оварда мешавад:

```
ad=&n;
```

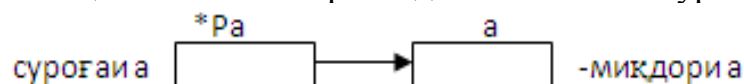
Дар ин чо маҳалли тағйирёбандаи n ба тағйирёбандаи ad бахшида шудааст. Бояд қайд кард, ки оператори ишорагар на қимати тағйирёбанда, балки суроғай онро муайян мекунад. Масалан, бигзор тағйирёбандаи a аз навъи бутун буда, тағйирёбандаи Pa аз навъи ишорагар бошад, яъне:

```
int a;
int *Pa;
```

Ҳоло метавонем бо ёрии оператори бахшиш суроғай a-ро ба тағйирёбандаи *Pa бахшид, яъне:

```
*Pa=a;
```

Тавре ки қайд намуда будем, тағйирёбандаи *Pa на қимат, балки танҳо суроғай a-ро нигоҳ медорад, аз ин рӯ, онро тағйирёбандаи ишорагар (pointer variable) меноманд. Алоқаи байни тағйирёбандаи a ва *Pa ба сурати зерин аст



Тағйирёбандаҳои a ва *Pa ҳар ду дар ҳофиза хонаҳои яксонро банд мекунанд. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    int a,b;
    int *Pa;
    cout<<"a=";<<cin>>a;
    Pa=&a; b=*Pa;
    cout<<"Addressi a="<<&a<<" qimati Pa="<<Pa<<endl;
    cout<<"Qimati a="<<a<<" qimati b="<<b<<endl;
```

```

cout<<"Munosibati * va & bo &*Pa="<<&*Pa<<"*&Pa="<<*&Pa<<endl;
return 0;
}

```

Тавре ки дида мешавад, дар ин чо тағйирёбандаи а ва b ҳар ду як миқдорро муаррифӣ мекунад. Бинобар ин, оператори * дар мавриди *Pa муҳтаво ё ҷойи тағйирёбандаи а-ро бармегардонад, ки суроғи он дар тағйирёбандаи Pa қарор дорад. Аз ин рӯ, онро тағйирёбандаи ғайримустақим низ меноманд. Натиҷаи барнома ба сурати зерин аст:

```

A=30
Addressi a=1245064      qimati Pa=1245064
Qimati a=30             qimati b=30
Munosibati * va & bo yakdigar &*Pa=1245064
*&=1245064

```

Ёфтган ё пайдо кардани суроғи як тағйирёбанда

Тавре ки гуфта гузаштем, ҳар тағйирёбанда дар ҳофиза вобаста ба навъаш (types) ҷойи муайнро ишғол мекунад, яъне дорои суроғи муайян аст. Баъзан барои ба он дастрас шудан беҳтар аст ба ҷойи худ он ба суроғи он муроҷиат кунем. Барои ба даст овардани суроғи тағйирёбанда аз оператори амперсанд «&» истифода бурда мешавад. Масалан, фарз мекунем, ки тағйирёбандаи x аз навъи double буда, дорои суроғи 124506 бошад. Ифодаи Px=&x миқдори 2345060 –ро дар тағйирёбандаи Px захира мекунад. Мисол:

```

#include <iostream>
using namespace std;
int main()
{
    double x,b;
    double *Px;
    cout<<"x=";  cin>>x;
    Px=&x;
    cout<<"Addressi x="<<&x<<endl;
    cout<<"Qimati x="<<x<<endl;
    cout<<"Qimati Px="<<Px<<endl;
    return 0;
}

```

Натиҷаи барнома ба сурати зерин аст:

```

X=3.14
Suragai
x=1245060
Qimati x=3.14
Qimati
Px=1245060

```

Аз матлаби баҳсамон чунин хулоса мебарояд, ки оператори «*» ба ду маҳфум ба кор бурда мешавад.

а) Барои муаррифӣ намудани тағйирёбандаҳои аз навъи ишорагар. Дар ин ҳолат дар тарафи чапи тағйирёбанда навишта мешавад. Ба монанди:

```
int *a, *b, *c;
float *x, *y;
char *s, *s1;
```

b) Барои дастёбди ба қимати тағйирёбандае, ки дар як ишорагар қарор дорад. **Мисол:**

```
#include <iostream>
using namespace std;
int main()
{
    char *P_chr;
    char chr1='a',chr2;
    cout<<"Qimati chr1=="<<chr1<<endl;
    cout<<"Surogai P_chr="<<&P_chr<<endl;
    P_chr=&chr1;
    cout<<"Qimati tagiiryobandae, ki surogai on dar P_chr qaror
dorad="<<*P_chr<<endl;
    chr2=*P_chr;
    cout<<"Qimati chr2=="<<chr2<<endl;
    return 0;
}
```

Дар ин барнома тағйирёбандаи P-chr аз навъи char (рамзи) буда, тағйирёбандаҳои chr1 ва chr2 низ аз навъи char мебошанд. Ҳангоми муаррифкунӣ ба тағйирёбандаи chr1 ба сифати қимати аввала рамзи 'а' бахшида шудааст. Дар оператору хуруҷии яқум , яъне

```
cout<<"Qimati chr1=="<<chr1<<endl;
```

Миқдори тағйирёбандаи chr1, ки дорои қимати аввала буд, ба чоп дода шудааст. Дар оператори хуруҷии дуввум суроғи тағйирёбандаи P-chr ба чоп дода шуда, пас аз он бо ёрии оператори ҷойгузорӣ, яъне

```
P_chr=&chr1;
```

Суроғи тағйирёбандаи chr1 ба ба он нисбат дода шудааст. Дар оператори хуруҷии саввум бошад, тағйирёбандаи P-chr, ки пештар бо ёрии оператори ҷойгузори, ки суроғи chr1 ба он нисбат дода шуда буд, бо ёрии оператори ишорагар ба чоп дода шудааст ва Тавре ки мебинем, миқдори chr1, яъне 'а' ба чоп дода шудааст. Баъдан ба ёрии оператори ҷойгузори миқдори P-chr ба тағйирёбандаи chr2 бахшида шуда, сипас бо ёрии оператори хуруҷи ба чоп дода шудааст.

Ба ғайр аз ин, агар дар суроғи ишорагарҳо миқдореро илова ва ё кам кунем, суроғи нав пайдо мешавад. Масалан, фарз мекунем, ки тағйирёбандаи ишорагари P ба қимати 300 баробар буда, барои навиштани навъи он 4 байт чудо карда шуда бошад. Ҳангоми ба P илова кардани ягон миқдори бутуни n , яъне P+n қимати P ба P+n*4 баробар мешавад. Бигзор, n=3 бошад, қимати P=312 мешавад. Аз ин ҷо шояд саволе ба миён ояд, ки чаро ба илова кардани 3 воҳид қимати P 12 воҳид зиёд мешавад?. Тавре ки пештар дар мавзӯи навъи додаҳо гуфта будем, ҳар навъи додаҳо дорои миқдори муайяне мебошанд. Аз ин ҷо ҳар гуна тағйирёбандаи навъи double дар ҳофиза 4 байтро ишғол мекунад.

Пас, миқдори иловашуда ба миқдори навъи дода зарб карда мешавад, яъне $12=3*4$

Инчунин аз як ишорагар ишорагари дигарро тарҳ кардан суроғи нав пайдо мешавад. Масалан, бигзор тағйирёбандаи ишорагари P дорои қимати 3000 буда, тағйирёбандаи ишорагари X дорои қимати 1000 бошад. Ифодаи $P-X=500$ мешавад, яъне $(3000-1000)/4=500$ (Дар ин ҷо 4 миқдори навъи дода мебошад).

Ба ишорагар бахшидани қимати аввала

Ба монанди дигар тағйирёбандаҳои муқаррарӣ ҳангоми муаррификунии тағйирёбандаҳои навъи ишорагар ба онҳо низ қимати аввали бахшида мешавад. Ишорагар метавонад ба сифати қимати аввала қимати Null ё суроғи ягон тағйирёбандаи қаблан муаррифшударо бахшид. Масалан :

```
int n;  
int *Pn=&n;
```

Бояд қайд кард, ки як тағйирёбанда пеш аз он ки муаррифӣ шавад, наметавонад барои тағйирёбандаи ишорагар қимати аввала бошад, аз ин ҷо ифодаҳои зерин ғайри қобили қабул аст.

```
int *Px=&x  
int x;
```

Дар забони C++ мафҳуми Null мавҷуд аст, ки онро доимии рамзики Null мегӯянд. Доимии мазкур дар чандин китобхонаҳо (Header Files) мавҷуд аст, бахусус дар китобхонаи <iostream>. Мисол:

```
char *x;  
x=0;
```

Ишорагари Null дар забони C++ истифодаи васеъ дорад, хусусан дар операторҳои идоракунӣ ё контролӣ (control-flow statements). Миқдори Null дар операторҳои мазкур ҳамчун false (ноҳақ) ва ҳар миқдори ғайринул ҳамчун true фаҳмида мешавад. Масалан, дар оператори даврии бо шарти пешоянд, то замоне ки шарти гузошташуда ба як ишорагари Null баробар шудан идома меёбад.

```
chr *P  
:  
while(p)  
{  
:  
}
```

Бояд қайд кард, ки барои ишорагарҳо низ метавон, амалҳои арифметикиро истифода кард. Ҳангоми зиёд ё кам кардани ишорагари ягон объект, ишорагари объекти дигар пайдо мешавад.

Мисоли 1. Ду тағйирёбандаи бутун муаррифӣ намуда, сипас, ҷойи қимати ин тағйирёбандаҳоро бо истифода аз ишорагарҳо иваз кунед.

```
#include <iostream>  
using namespace std;  
int main() {
```

```

int a,b;
cout<<"a=";cin>>a;
cout<<"b="; cin>>b;
int* p=new int;
*p=a;
a=b;
b=*p;
delete p;
cout<<"a="<<a<< " b="<<b;
return 0;
}

```

Мисоли 2. Як ишорагари типи бутун муаррифӣ намуда, сипас барои он ҳофиза чудо намоед. Барои ин ишорагар аз клавиатура қимат ворид намуда, пас аз он қимати онро ду воҳид зиёд кунед. Натиҷаи ҳосилшударо хориҷ намоед.

```

#include <iostream>
using namespace std;
int main() {
    int *p;
    p=new int;
    cin>>*p;
    (*p)+=2;
    cout<<*p;
    delete p;
    return 0;
}

```

Даъвати функсия аз рӯи суроға ва баргардонидани қимат

Дар аксар забонҳои барноманависӣ ду роҳ барои раван кардани аргументҳо ба функсия ё даъвати аргументҳо вуҷуд дорад:

1. Даъват ба миқдор (call by value)
2. Даъват ба нишона (call by reference)

Ҳангоми даъват намудан ба миқдор - тавре ки дар мавзӯи функсияҳо мушоҳида ва истифода кардем, ҳамаи миқдори дода нусхабардорӣ шуда, ба функсия равона карда мешавад. Тағйир додани нусха ғайриимкон аст, зеро қимати асли (original) дар функсияи даъват кунанда мавҷуд аст. Ҳангоми нусхабардорӣ намудан агар додаҳо сатрҳо сохторҳо ва массивҳо бошанд, вақти зиёд сарф мешавад. Аз ин рӯ, даъват ба нишона (address) истифода карда мешавад. Даъват ба суроға имконияти мустақим дастрас шуданро ба додаҳо дода, инчунин имконият медиҳад, миқдори дода тағйир дода шавад.

Мисол. Барномае нависед, ки та-йирёбандаеро ба миқдор ба функсия равона кунад.

```

#include <iostream>
using namespace std;
void gets2(int , int);
int main(){
    int x,y;
    cout<<"x=";cin>>x;
}

```

```

    cout<<"y=";cin>>y;
    cout<<"Qimathoi x va y pesh az da`vati function-i far`i"<<endl;
    cout<<"x="<<x<<"\ty="<<y<<endl;
    gets2(x,y);
    cout<<"Qimathoi x va y pas az da`vati function-i far`i"<<endl;
    cout<<"x="<<x<<"\ty="<<y<<endl;
    return 0;
}
void gets2(int x, int y)
{
    x+=2;
    y+=2;
    cout<<"Qimathoi x va y dar function-i far`i"<<endl;
    cout<<"x="<<x<<"\ty="<<y<<endl;
}

```

Қиматҳои а ва b дар ба функсияи фаръӣ равои карда мешаванд ва сипас дар он чо бо ёрии операторҳои хуруҷӣ ба ҳар кадоми он миқдори ба 2 воҳид баробар илова намуда, ба чои дода мешаванд, сипас дар функсияи аслии функсияи фаръӣ даъват карда мешавад ва бо ёрии оператори хориҷкунӣ миқдори тағйирёбандаҳои мазкур ба чои дода мешавад. Тавре ки дида мешавад, дар функсия аслии амале, ки дар функсияи фаръӣ иҷро шуда буд ҳеҷ таъсире нагузошт. Натиҷаи барнома ба сурати зерин аст:

```

x=5
y=3
Qimathoi x va y pesh az da`vati function-i far`i
x=5      y=3
Qimathoi x va y dar function-i far`i
x=7      y=5
Qimathoi x va y pas az da`vati function-i far`i
x=5      y=3

```

Дар фуроҳонӣ бо суроға бошад, ба сифати қимат суроғаи дода ба функсияи фаръӣ равои карда мешавад ва ҳар амале, ки дар функсияи аслии дар болои он дода анҷом дода мешавад, дар функсияи аслии низ таъсир мегузорад. Ҳангоми ба таври мазкур равои намудани аргументҳо ба функсияи фаръӣ дар пешнамунасозии функсияи фаръӣ пас аз навъи аргумент, оператори амперсант (&) навишта мешавад.

Мисол. Мисоли дар боло овардашударо ба таври даъват аз рӯи нишона менависем.

```

#include <iostream>
using namespace std;
void gets2(int &, int &);
int main(){
    int x,y;
    cout<<"x=";cin>>x;
    cout<<"y=";cin>>y;
    cout<<"Qimathoi x va y pesh az da`vati function-i far`i"<<endl;
    cout<<"x="<<x<<"\ty="<<y<<endl;
}

```

```

    gets2(x,y);
    cout<<"Qimathoi x va y pas az da`vati function-i far`i"<<endl;
    cout<<"x="<<x<<"\ty="<<y<<endl;
    return 0;
}
void gets2(int &x, int &y)
{
    x+=2;
    y+=2;
    cout<<"Qimathoi x va y dar function-i far`i"<<endl;
    cout<<"x="<<x<<"\ty="<<y<<endl;
}

```

Натиҷаи барнома ба сурати зерин аст:

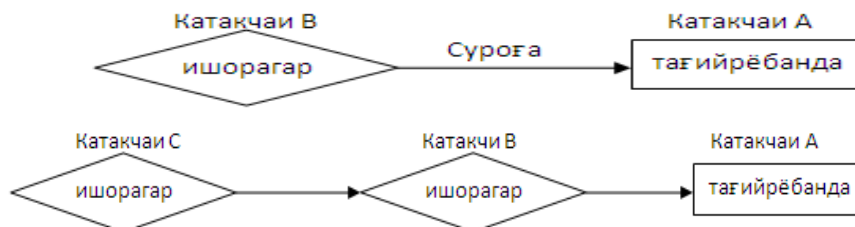
```

x=5
y=3
Qimathoi x va y pesh az da`vati function-i far`i
x=5      y=3
Qimathoi x va y dar function-i far`i
x=7      y=5
Qimathoi x va y pas az da`vati function-i far`i
x=7      y=5

```

Ишорагар бар ишорагар (Pointer to Pointer)

Чи хеле ки қаблан ишора гардид, ишорагар низ тағйирёбанда буда, қимати он адади додашуда набуда, балки суроғаи тағйирёбандаи дигар мебошад. Аз ин ҷо метавонем ишорагарро чунин таъриф диҳем: тағйирёбандае, ки дар худ суроғаи тағйирёбандаи дигарро нигоҳ медорад, ишорагар номида мешавад. Инчунин метавонем ишорагари дигарро бар ишорагар муаррифӣ кунем, ки онро ишорагар бар ишорагар (pointer to pointer) ё **MULTIPLE INDIRECTION** меноманд. Дар поён ба намуди диаграмма ду навъи ишорагар ишорагар бар тағйирёбандаи оддӣ ва ишорагар бар ишорагар нишон дода шудааст.



Тавре ки мушоҳида мешавад, ишорагари якум дар худ суроғаи ишорагари дигарро дорад, он ҳам бошад, дар навбати худ суроғаи тағйирёбандаи дигарро нигоҳ медорад. Ин равиш метавонад якчанд маротиба иҷро гардад. Барои муаррифӣ намудани тағйирёбандаи ишорагар бар ишорагар пеш аз номи тағйирёбанда ду оператори «*» гузошта мешавад. Наҳви таърифи ишорагар бар ишорагар ба сурати зерин аст:

```
data_type **Pvariable
```

Дар ин чо навъи тағйирёбанда, Ptvariable-номи тағйирёбанда ва ва ду рамзи «*» бошад маънои ишорагар бар ишорагарро доранд. Масалан, тағйирёбандаи бутуни x аз навъи ишорагар бар ишорагари чунин int **x муаррифӣ карда мешавад. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    int n,*x,**y;
    cout<<"n=";<<cin>>n;
    x=&n;
    y=&x;
    (*x)++;
    (**y)--;
    cout<<"n="<<n<<"\n *x="<<*x<<"\n **y="<<**y<<"\n x="<<x<<endl;
    return 0;
}
```

Тавре ки дар барнома дида мешавад, пас аз муаррифӣкунии тағйирёбандаҳо бо ёрии оператори дохилкунӣ қимати n ворид карда мешавад, сипас, ба ёрии оператори ҷойгузори суроға n ҳам ба *x ва ҳам ба тағйирёбандаи **y бахшида шудааст. Сипас, қиматҳои ҳам *x ва ҳам **y 5 вохид афзоиш дода шудааст. Дар охир ба ёрии оператори хоричкуни қиматҳои *x ва **y инчунин бар тағйирёбандаҳои мазкур бори дигар ишорагар таъсис дода шуда ба чоп бароварда мешаванд. Натиҷаи барнома ба сурати зерин аст:

```
N=50
N=60
*x=60
**y=60
X=1245064
Y=1245060
```

Аз натиҷаи барнома бармеояд, ки ҳангоми суроғаи тағйирёбандаро афзоиш додан, қимати он ҳам афзоиш дода мешавад, миқдори афзоишшуда ба миқдори ҳофизае, ки компилятор барои навъи дода ҷудо мекунад, зарб карда мешавад. Аз ин ҷост ки қимати n=60 мешавад.

Истифодаи ишорагарҳо барои кор кардан бо массивҳои дученака

Монанди массивҳои якченака метавон массивҳои дученакаро низ ба таври динамикӣ муаррифӣ кард. Дар ин сурат аввал массиви дученака ҳамчун як ишорагари оддӣ муаррифӣ карда шуда, сипас, барои он бо андозаи MxN (Дар ин чо M-миқдори сатрҳо ва N- миқдори сутунҳои массив мебошад.) тавассути операторҳои кор бо ҳофизаи динамикӣ ҳофиза ҷудо карда мешавад. Барои ба массивҳои дученака ҷудо намудани ҳофизаи динамикӣ аз яке аз роҳҳои зерин истифода бурда мешавад.

1. Тавассути функсияи calloc (). Наҳви истифодаи функсияи мазкур барои массивҳои дученакаи динамикӣ намуди зеринро дорад:

```
int *a, N, M;
.....
```

```
a=(int *) calloc(M*N, sizeof(int));
```

2. Тавассути функцияи malloc. Наҳви истифодаи функцияи мазкур барои массивҳои дученака намуди зеринро дорад:

```
int *a, N, M;
```

```
.....
```

```
a=(int *) malloc(M*N*sizeof(int));
```

3. Тавассути функцияи аз оператори new. Наҳви истифодаи оператори мазкур барои массивҳои дученака намуди зеринро дорад:

```
int *a, N, M;
```

```
.....
```

```
a=new int[M*N];
```

Дар ин ҷо дар ҳама операторҳо элементҳои матритса ҳамчун массиви якченака захира карда мешаванд. Дар ин массиви якченака аввал элементҳои сатри якум, баъд дуюм ва ҳоказо илова карда мешаванд. Барои ба элементи $a[i][j]$ муроҷиат кардан лозим аст, ки тавассути рақами сатр i ва рақами сутун j рақами элементи k -ро ҳисоб кард.

Фарз мекунем, ки элементҳои массиви дученака шуруъ карда, аз 0-рақамгузори шуда бошанд, пас $k=i*M+j$ мешавад. Барои ба элементи $a[i][j]$ -дастрас шудан аз оператори зерин истифода бурда мешавад:

```
*(a+i*N+j);
```

Мисол. Массиви динамикии андозааш $M*N$ дода шудааст. Барномае нависед, ки элементи хурдтарини матритсаро ҳисоб кунад.

```
#include <iostream>
using namespace std;
int main(){
    int *a, M, N, i, j, min;
    cin>>M>>N;
    a=(int *) calloc(M*N, sizeof(int));
    for (i=0; i<M; i++)
        for (j=0; j<N; j++)
            cin>>*(a+i*N+j);

    min=*a;
    for (i=0; i<M; i++)
        for (j=0; j<N; j++)
            if (min>*(a+i*N+j))
                min=*(a+i*N+j);

    cout<<min<<endl;
    return 0;
}
```

Тавре ки аз ин ҷо дида мешавад, пас аз муаррификунии ишорагар (массиви дученака) андозаи он тавассути оператори дохилкунӣ ворид карда шудааст. Пас аз дохил намудани андозаи массив, барои он ҳофизаи лозимӣ ҷудо карда шуда, элементҳои он дохил карда шудаанд. Пас аз дохил намудани элементҳои массиви дученака, элементи якум ба намуди пешфарз min ҳисоб карда шуда, сипас дар дохили оператори такрорӣ бо дигар элементҳо муқоиса карда шудааст.

Операторҳои дигар низ монанди ин истифода карда мешаванд.

Истифодаи ду ишорагар барои массивҳои дученакаи динамикӣ

Массивҳои дученакаи динамикиро метавон низ ба сурати ду ишорагар, ки ишорагар ба ишорагар номида мешавад, эълон кард. Наҳви истифодаи ин амал намуди зеринро дорад:

```
int **a, m, n;  
....  
a=new int *[m];
```

Дар ин ҷо массиви дорой n -элемент сохта шудааст, ки ҳар як элементи он суроға буда, ба суроғаи сабтшудаи `int` ишора мекунад. Барои ишорагарҳои боқимонда ҳалкаи такрорие аз 0 то $n-1$ ташкил карда мешавад, ки дар ҳар кадом ишорагар суроғаи хотирае, ки дар он ҷо m -элементи навъи `int` нигоҳ дошта мешавад. **Мисол:**

```
for (i=0; i<n; i++)  
    a[i]=new int [m];
```

Баъд аз ин массиви дорой n ишорагар муаррифӣ карда мешавад, ки дар ҳар як ишорагар суроғаи массиви дорой m -элемент нигоҳ дошта мешавад. Мурочиат ба элементҳои ин навъ массивҳои дученака монанди мурочиат ба массивҳои дученакаи оддӣ мебошад.

Мисол. Барномае нависед, ки ҳосили зарби ду массиви дученакаро ҳисоб кунад. Аз нуқтаи назари математикӣ формулаи зарби ду матрица намуди зеринро дорад.

$$c_{ij} = \sum_{k=0}^{m-1} a_{ik} * b_{kj}$$

Матни барнома:

```
#include <cstdlib>  
#include <iostream>  
using namespace std;  
int main(){  
    int i,k ,j ,l , m, n, **a, **b, **c;  
    cin>>m>>n>>l;  
    a=new int *[n];  
    for (i=0; i<m; i++)  
        a[i]=new int [m];  
    cout<<"\n Enter Matrix one \n";  
    for (i=0; i<m; i++)  
        for (j=0; j<n; j++)  
            cin>>a[i][j];  
    for (i=0; i<m; cout<<endl, i++)  
        for (j=0; j<n; j++)  
            cout<<a[i][j]<<"\t";  
    cout<<"\n Enter Matrix two\n";  
    b=new int *[l];  
    for (i=0; i<n; i++)  
        b[i]=new int [n];  
    for (i=0; i<n; i++)  
        for (j=0; j<l; j++)
```

```

        cin>>b[i][j];
c=new int *[l];
for (i=0; i<m; i++)
    c[i]=new int [m];
for (i=0; i<m; i++)
    for (j=0; j<l; j++)
        for (c[i][j]=0, k=0; k<n; k++)
            c[i][j]+=a[i][k]*b[k][j];
for (i=0; i<n; cout<<endl, i++)
    for (j=0; j<l; j++)
        cout<<c[i][j]<<"\t";
for (i=0; i<n; i++)
    delete [] a[i];
delete [] a;
for (i=0; i<l; i++)
    delete [] b[i];
delete [] b;
for (i=0; i<l; i++)
    delete [] c[i];
delete [] c;

return 0;
}

```

Барои боз ҳам пурратар шинос шудан мисоли дигареро дида мебароем.
Мисол. Матритсае тартиб дихед, ки элементҳои он аз ададҳои тасодуфи иборат бошанд, сипас элементҳоро хориҷ кунед.

```

#include <cstdlib>
#include <iostream>
using namespace std;
int main(){
    int **p;
    int m,n,i,j;
    cout<<"n=";cin>>n;
    cout<<"m=";cin>>m;
    p=new int *[n];
    for (i=0;i<n;i++)
    {
        p[i]=new int[m];
        for(j=0;j<m;j++) {
            p[i][j]=rand()%50;
            cout<<p[i][j]<<" ";
        }
        cout<<endl;
    }
    for (i=0;i<n;i++)
        delete [] p[i];
    delete []p;
    return 0;
}

```


Тавре ки аз мантиқи барнома дида мешавад, ки дар ин ҷо тағйирёбандаи `p` ҳамчун ишорагар бар ишорагар тавсиф шудааст, ин нишон медиҳад, ки `p` массиви дученака тавсиф шудааст. Баъд ки-матҳои сатр ва сутун аз клавиатура бо ёрии истифодабаранда ворид карда мешаванд, ифодаи `p=new int *[n];` нишондиҳандаи массив аз ишорагар мебошад. Баъдан дар қисмати зерин дар дохили ду оператори даврӣ аз ададҳои тасодуфӣ элементҳои массив пур карда шуда, ба ҷоп дода шудаанд.

```

for (i=0;i<n;i++){
    p[i]=new int[m];
    for(j=0;j<m;j++){
        p[i][j]=rand()%50;
        cout<<p[i][j]<<" ";
    }
    cout<<endl;
}

```

Дар қисмати зерин дар дохили як давр бо ёрии оператори `delete` хотира аз массиви якченакаи динамикӣ озод карда шуда, дар охир бо ёрии фармони `delete []p` ҷойи хотираи асосии низ аз массив озод карда шудааст.

```

for (i=0;i<n;i++)
    delete [] p[i];
delete []p;

```

Массивҳо ва ишорагарҳо

Метавонем ҳангоми муаррификунии массивҳо онҳоро аз навъи ишорагарҳо таъриф кунем, яъне элементҳои массивро аз ишорагарҳо (ишорагаре, ки суроғи тағйирёбандаи додасударо дар худ нигоҳ ме-дорад) пур кунем.

```
int *ar[10]
```

Масалан, барои суроғи панҷуми массивро аз ягон тағйирёбандаи `z` пур кунем, менависем

```
*ar[2]=&z
```

Барои ба даст овардани қимати `z` аз ишорагари дукарата, яъне ишорагар бар ишорагар истифода бурда мешавад, ба монанди

```
**x[2];
```

Дар забони C++ миёни ишорагарҳо ва массивҳо робитаи наздике вучуд дорад. Медонем, ки номи массив, суроғи элементи массив мебошад. Бигзор массиви бутунӣ `b[5]` ва тағйирёбандаи ишорагари `Pb` мавҷуд бошад. Бинобар ин, ду оператори зерин ба ҳам эквивалент мебо-шанд.

```
Pb=b ⇔ &b[0]
```

```
char str[80],*P;
```

```
P=str;
```

```
P=&str[0] ⇔ P=str
```

Дар қитъаи барномаи боло дида мешавад, ки `P` суроғи элементи массивро нишон медиҳад. Агар хоҳем, ки ба элементи чоруми массив дастрас шавем, аз операторҳои зерин истифода мекунем:

```
str[3]   ё   *(str+3)
```

Ҳар ду оператор элементи чоруми массивро нишон медиҳанд. Ба таври умуми ғӯем, операторҳои $p[n]$ ва $*(p+n)$ аъзои чоруми массивро нишон надода, элементи $(n+1)$ -ро нишон медиҳанд, зеро ки зернависи якуми массив нол мебошад. Аз ин матлаби баҳсшуда бармеояд, ки элементҳои массивро ҳам аз рӯи зернавис ва ҳам аз рӯи суроға дастрас кардан мумкин аст, вале дастрасшавӣ аз рӯи суроға нисбат ба зернавис саҳеҳтар мебошад. Ба таври умумӣ ба элементҳои массив метавон бо роҳҳои гуногун дастрас шуд.

Мисол. Барномае нависед, ки ба элементҳои массив бо роҳҳои гуногун дастрас гардиданро амалӣ гардонад.

```
#include <cstdlib>
using namespace std;
int main(){
    const int n=5;
    int b[]={5,8,7,62,55};
    int *Pb=b;
    int i;
    cout<<"Ba namudi muqarrari dastras shudan"<<endl;
    for (i=0;i<n;i++)
        cout<<"b["<<i<<"]="<<b[i]<<endl;
    cout<<"Ba vositai ishoragar dastras shudan"<<endl;
    for (i=0;i<n;i++)
        cout<<"*(b+"<<i<<"]="<<*(b+i)<<endl;
    cout<<"az rui Address dastras shudan"<<endl;
    for (i=0;i<n;i++)
        cout<<"Pb["<<i<<"]="<<Pb[i]<<endl;
    cout<<" Ba vositai ishoragar bar Address dastras shudan "<<endl;
    for (i=0;i<n;i++)
        cout<<"*(Pb+"<<i<<"]="<<*(Pb+i)<<endl;
    return 0;
```

Миқдори тағйирёбандаҳои ишорагар наметавонанд ба таври дилхоҳ тағйир ёбанд. Аз ин рӯ, истифодаи оператори як воҳид афзоиш (increment) ва як воҳид коҳиш (decrement) барои элементҳои массив маслиҳат дода намешавад. Тавре ки дар гузашта дида будем, вобаста аз навъи дода дар хотира ҷойи махсусро ишғол мекунад. Ба-рои миқдори навъҳои стандартиро мушаххас намудан аз оператори ҳофиза sizeof истифода бурда мешавад.

Мисол. Барномае нависед, ки миқдори ҳофизаи ҳамаи навъҳои стандартиро барорад.

```
#include <iostream>
using namespace std;
int main(){
    char chr;
    short s;
    int i;
    long l;
    float f;
    double d;
    long double ld;
    int array[20],
```

```

        *Parray=array;
        cout<<"Miqdori nav`hoi standarti"<<endl;
        cout<<" sizeof char="<<sizeof(chr)<<"\n sizeof
short="<<sizeof(s)<<"\n sizeof int="<<sizeof(i)<<"\n sizeof
long="<<sizeof(l)<<"\n sizeof double="<<sizeof(d)<<"\n sizeof long
double="<<sizeof(ld)<<"\n sizeof array=" <<sizeof(Parray) <<endl;
        return 0;
}

```

Оператори sizeof имконияти қабули дилхоҳ тағйирёбанда навъи стандартӣ ва худи навъи стандартиро дорад. Бояд қайд кард, ки sizeof функсия набуда, оператори ягона мебошад. Натиҷаи барнома ба сурати зерин аст:

```

Miqdori nav`hoi
standarti
sizeof char=1
sizeof short=2
sizeof int=4
sizeof long=4
sizeof double=8
sizeof long double=10
sizeof array=4

```

Равон кардани массив ба функсия ба сифати аргумент

Барои массивро ба функсия ба сифати аргумент раво кардан, фақат суроғаи массив ба функсия интиқол дода мешавад, ни нусхаи тамоми элементҳои массив, пас аз ин ҷо барои массивҳо фурӯхонӣ ба миқдор (call byvalue) ба кор намеояд. Ҳангоми дар функсия ба сифати аргумент даъват кардани массив як ишорагар бар элементи аввалии массив ба функсия равона карда мешавад. Номи массив бе зернавис як ишорагар ба элементи якуми массив аст. Умуман, се механизм (роҳ)-и раво кардани массив ба сифати параметри функсия мавҷуд аст.

1. Параметри функсия дар ин равиш ба намуди массив муаррифӣ мегардад. Ба монанди мисоли зерин:

```

#include <iostream>
using namespace std;
void show( int a[25]){
    for (int i=0;i<25;i++)
        cout<<a [i]<<"\t";
    cout<<endl;
}
int main(){
    int a[25],i;
    for (i=0;i<25;i++)
        a[i]=rand()%65;
    show(a);
    return 0;
}

```

Дар ин мисол функсияе ба номи show муаррифӣ шуда аст, аргументи он массиви дорои 25-элемент мебошад. C++ онро ҳамчун як ишорагар навъи бутун (integer pointer) табдил медиҳад, зеро ҳеҷ параметр наметавонад, ки тамоми элементҳои массивро дарёфт кунад.

2. Дар ин намуд ба сифати аргумент массив бе нишон додани андозааш навишта мешавад. **Мисол:**

```
#include <iostream>
using namespace std;
void show( int a[]){
    for (int i=0;i<25;i++)
        cout<<a [i]<<"\t";
    cout<<endl;
}
int main(){
    int a[25],i;
    for (i=0;i<25;i++)
        a[i]=rand()%65;
    show(a);
    return 0;
}
```

Тавре ки дида мешавад, дар ин намуд андозаи массив муайян нашудааст. Дар ин ҷо низ a[]-як ишорагар ба элементи якуми массив мебошад.

3. Дар ин равиш як ишорагаре ба номи массив муаррифӣ карда шуда, ба сифати аргументи функсия даъват карда мешавад. **Мисол:**

```
#include <iostream>
using namespace std;
void show( int *a,int n)
{
    for (int i=0;i<n;i++)
        cout<<a [i]<<"\t";
    cout<<endl;
}
int main(){
    int a[25],i;
    for (i=0;i<25;i++)
        a[i]=rand()%65;
    show(a,5);
    return 0;
}
```

Дар ин мисол ба сифати аргументи функсия ду адади бутун истифода шудааст, ки яке аз онҳо ишорагар ба массив буда, дигараш бошад, андозаи массив ба ҳисоб меравад. Як чизро дар хотир бояд дошт, ки ҳангоме, ки массив ба сифати аргументи функсия тавсиф мегардад, ҳамчун як ишорагар фаҳмида мешавад. Ҳар амалиёте, ки болои массив дар функсияи фаръӣ иҷро карда мешавад, дар функсияи аслий низ болои элементҳои массив низ амалӣ мегардад.

Мисол. Функсияе нависед, ки элементҳои массиви додашударо аз рӯи алгоритми мураббабсозии ҳубобӣ мураббаб созад.

```

#include <iostream>
using namespace std;
void bubblesort( int *,const int);
int main(){
    const int n=25;
    int a[n],i;
    cout<<"Elementhoi massiv pesh az jobajoguzory\n"<<endl;
    for (i=0;i<n;i++)
    {
        a[i]=rand()%65;
        cout<<a [i]<<"\t";
    }
    cout<<endl;
    cout<<"\nElementhoi massiv pas az jobajoguzory\n"<<endl;
    bubblesort(a,n);
    for (i=0;i<n;i++)
        cout<<a [i]<<"\t";
    cout<<endl;
    return 0;
}
void bubblesort( int *massiv, int n)
{
    int temp,i,j;
    for (i=0;i<n;i++)
        for(j=0;j<n-i;j++)
            if (*(massiv+j)>*(massiv+j+1))
            {
                temp=*(massiv+j);
                *(massiv+j)=*(massiv+j+1);
                *(massiv+j+1)=temp;
            }
}

```

Тавре ки дида мешавад, функсияи фаръӣ дорои ду аргумент мебошад, ки якеаш ишорагар бар массив буда, дигараш нишон-диҳандаи андозаи массив мебошад.

Ҳангоми равоӣ кардани массиви n-ченака дар функсия ба ғайр аз андозаи яқум, андозаи боқимондаҳо нишон дода мешаванд. Дар массивҳои якченака ҳангоми равоӣ кардани он ба функсия андозааш нишон дода намешавад (равиши дуюм). Массивҳои n-ченака бошанд (сар карда аз дученака), массив бар массив мебошанд. Дар ин навъ массивҳо бошад, элементҳои яқум ҳудаш ҳам массив мебошад. Барои элементҳои баъдина бошад, бояд андоза мушаххас гардад, то ин ки компилятор барои он дар ҳофиза ҷоеро банд намояд. Мисол :

```

#include <iostream>
using namespace std;
void show2( int a[][3],int n)
{
    int i,j;
    for (i=0;i<n;i++) {
        for ( j=0;j<3;j++){
            cout<<a [i][j]<<" ";

```

```

    }
    cout<<endl;
}
}
int main()
{
    int a[3][3]={
        {5,4,7},
        {1,0,22},
        {7,-8,3},
    };
    show2(a,3);
    return 0;
}

```

Дар ин мисол ишораг бар элементи дуввуми массив ишора кардааст.

Мисоли дигар. Массиви а, ки дорои n элемент мебошад, дода шудааст, функсияе нависед, ки элементҳои чуфти массивро барорад.

```

#include <iostream>
using namespace std;
void Print(int *a,int n){
    for (int i=0;i<n;i++)
        if ((a[i] %2) ==0)
            cout<<a[i]<<"\t";
    cout<<endl;
}
int main() {
    const int n=50;
    int a[n],i;
    for (i=0;i<n;i++)
    {
        a[i]=rand()%25;
        cout<<a[i]<<"\t";
    }
    cout<<endl<<endl;
    Print(a,n);
    return 0;
}

```

Аз функсия баргардонидани суроға

То ҳол функсияҳое, ки мо навиштем, танҳо қимати ягон навъи стандартиро бармегардониданд. Илова бар ин функсия метавонанд ишорагарҳоро низ баргардонад. Барои иҷрои ин амал хангоми пешнамунасоӣ пеш аз номи функсия оператори ишорагар навишта мешавад. Ба барномаи зерин диққат диҳед:

```

#include <iostream>
using namespace std;
int *ishoragar( int &a){
    return &a ;
}

```

```

int main(){
    int a, *P;
    cout<<"a=";cin>>a;
    P=ishoragar(a);
    cout<<"Enter a="<<a<<endl;
    (*P)++;
    cout<<"Result a="<<a<<endl;
    return 0;
}

```

Тавре ки мушоҳида мешавад, дар ин ҷо функсияи ishoragar дорои як аргументи суроға мебошад. Ва ҳангоми фуруҳонӣ дар функсияи асли тавассути тағйирёбандаи ишорагари P суроғаи он баргардонида мешавад. Сипас, бо ишорагари P амали як воҳид афзоиш иҷро гардида, қимати a ба ҷоп дода мешавад. Натимча шакли зеринро дорад:

```

Enter a =5
Result a=6

```

Ба ғайр аз он ки дар боло зикр кардем, функсияҳо низ метавонанд ба сифати аргумент суроғагархоро баргардонанд. Барои иҷрои ин амал ҳангоми пешнамунасосӣ пеш аз номи функсия оператори амперсанд навишта мешавад. Барои ин ба мисоли зерин тавачҷуҳ фармоед:

```

#include <iostream>
using namespace std;
int &nishoragar( int &a){
    return a ;
}
int main(){
    int a,P;
    cout<<"Enter a=";cin>>a;
    P=ishoragar(a);
    P++;
    cout<<"Result a="<<a<<endl;
    cout<<"Result P="<<P<<endl ;
    return 0;
}

```

Натиҷаи барнома шакои зеринро дорад:

```

Enter a =5
Result a=6
Enter P =5

```

Пас аз он, ки ба моҳияти ишорагарҳо шинос шудам, як микросхемаи компютерро мавриди баҳс қарор медиҳем.

Тачҳизоти хотираи ғаврӣ

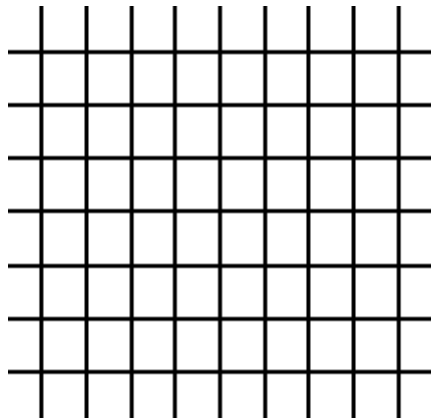
Ҳар барномае, ки мо менависем, ҳангоми иҷроиш дар хотираи ғаврӣ ҷой дода мешавад. Аз ин ҷо бо роҳи мустақим ё тавассути кэш протсессор фармонҳои барномаро хонда, иҷро мекунад. Барои ҳар барномасоз лозим аст, ки аз тарзи кори хотираи ғаврӣ хабардор бошад, то битавонад, барномаҳои лозимиро ба таври сарътар нависад.

Хотираи фаврӣ ин платаяи на онқадар калони микросхемавӣ мебошад, ки бо разъёмҳои махсус ба платаяи модари пайваस्त карда мешавад. Шакли умумии таҷҳизоти хотираи фаврӣ чунин аст:



Баъзан хотираи фавриро хотираи динамикӣ низ мегӯянд. Сабабаш дар он аст, ки нисбат ба дигар нигоҳдорандаҳо (ё интиқолдиҳандагон)-и иттилоот (диски дурушт, флешка ва ғайра) маълумотҳоро зудтар мехонад. Яке аз фарқияти дигари хотираи фаврӣ аз диски дурушт дар он аст, ки хотираи фаврӣ ба барқ вобаста мебошад. Аз ин ҷост, ки ҳангоми хомуш гардидани барқ маълумотҳои он нест карда мешаванд.

Пеш аз мавриди баҳс қарор додани таҷҳизоти мазкур, хотирнишон мекунем, ки воҳиди хурдтарини ахбор бит мебошад. Як бит ин қимати 0 ё 1 мебошад. Асоси хотираи фавриро катакчаҳои ба андозаи як бит ташкил медиҳанд, ки шакли матритсаро доранд. Барои навиштани як байт маълумот дар хотираи фаврӣ 8 бит катакча лозим меояд. Намуди умумии катакчаҳои хотираи фаври шакли зеринро доранд:



Тавре ки аз ин расм мушоҳида мешавад, катакча воҳиди минималии хотира буда, ба 1 бит баробар мебошад. Барои дар хотираи фаврӣ нигоҳ доштани 1 байт 8 катакчаи пайиҳам лозим меояд.

Тавре ки медонем, як барномаи навишташуда метавонад, додаҳои навъҳои гуногунро қорқард кунад. Барои нигоҳ доштани додаҳои ҳар навъ дар хотира минтақае ҷудо карда шуда, сипас амали лозими болои додаҳо иҷро карда мешавад. Барои намуна барномаи кучаки зеринро дида мебароем.

```
#include <iostream>
using namespace std;
int main(){
    int a=5;
    cout<<"Size: "<<sizeof(a)<<endl;
    cout<<"Value: "<<a<<endl;
    cout<<"Address: "<< &a<<endl;
    return 0;
}
```


Дар ин барнома тағйирёбандае аз навъи `int` эълон карда шуда, ба сифати қимати ибтидои барои он қимати 5 бахшида шудааст. Пас аз муаррификунӣ амалҳои зерин иҷро карда шудаанд:

- 1) Муайян намудани андозаи хотира;
- 2) Хориҷ намудани қимати тағйирёбанда;
- 3) Муайян намудани суроғаи тағйирёбанда.

Барои иҷрои ин амалҳо аз се сатри барнома, ки шакли зеринро доранд, истифода карда шудааст.

```
cout<<"Size: "<<sizeof(a)<<endl;
cout<<"Value: "<<a<<endl;
cout<<"Address: "<< &a<<endl;
```

Пас аз иҷроиши барнома натиҷа ба сурати зерин пешниҳод карда мешавад:

```
Size:4
Value: 5
Address: 1245008
```

Тавре ки дида мешавад, дар сатри якум андозаи тағйирёбандаи навъи `int`, дар сатри дуввум қимати тағйирёбанда ва дар сатри сеюм суроғаи тағйирёбанда хориҷ карда шудааст.

Аз ин ҷо шояд дар зеҳни хонандагон чунин савол пайдо шавад: Барои чӣ тағйирёбандаи навъи `int` 32 (4*8) катакчаи хотираи фавриро банд мекунад, лекин дорои як суроға аст?. Ҷавоби ин суол чунин аст: Суроғаи тағйирёбанда дар хотираи фаврӣ ин катакчаи нахустини хотира аст. Протсессор суроғаи нахустин ва андозаи навъи тағйирёбандаро доништа, медонад, ки барои он чӣ қадар катакча ҷудо кунад.

Тавре ки медонем, ҳамаи додаҳо дар хотира ба намуди дӯи нигоҳ дошта мешаванд, аз ин ҷо намуди дуии адади 1000 шакли 11 1110 1000 ро дорад. Аз ин ҷо он 10 катакчаро аз 32 катакчаи ҷудокардашуда банд мешаваду халос. Бинобар ин маслиҳат дода мешавад, ки аввал дарозии додаҳои лозимиро ба назар гирифта баъд навъи он муайян карда шавад. Масалан агар пешакӣ муайян бошад, ки қимати тағйирёбанда аз фосилаи -32768 ... 32767 берун намебарояд, аз навъи `short` истифода кардан хубтар мебошад, зеро он дар хотира 2 байт ҷойро банд мекунад.

Саволҳо барои мустаҳкамкунӣ

1. Ишорагар чист ва чӣ зарурати истифода дорад?
2. Чӣ тавр ишорагар таъриф карда мешавад?
3. Байни оператори ҳофиза ва ишорагарҳо чӣ хел робита мавҷуд аст?
4. Чӣ тавр ишорагар ба функсия равона карда мешавад?
5. Оё бо операторҳо амалҳои ҷамъ ва тарҳ ҷой дорад?
6. Ишорагар бар ишорагар гуфта чиро мефаҳмед?
7. Чӣ тавр суроғаи як тағйирёбандаро пайдо кардан лозим аст?
8. Ду ишорагар чиро ифода мекунад, ва барои чӣ лозим аст?
9. Оё метавон ба ишорагар низ қимати аввала бахшид?
10. Барои раво кардани аргумент ба функсия чанд роҳ мавҷуд аст?

11. Дар байни массивҳо ва ишорагарҳо ҷиробита вучуд дорад?
12. Чанд тарзи даъвати массив аз функсияи фаръӣ ва ё равон кардани элементҳои массив ба функсияи фаръӣ чанд роҳ мавҷуд аст?
13. Ба чанд тарз ба элементҳои массив дастрас шудан мумкин аст?
14. Оператори ҳофиза чӣ корро иҷро мекунад?
15. Тағйирёбандаҳои сатри чӣ гуна тағйирёбандаанд?
16. Таҷҳизоти хотираи фаврӣ чӣ гуна таҷҳизот аст?
17. Оё ба барномасоз лозим аст, ки дарозии доаҳои мавриди назарашро донанд?
18. Ҳангоми аз навъи `int` муаррифӣ намудани матритсаи `a[6][5]` чӣ қадар ҳофиза лозим мешавад?
19. Массивҳои бисёрченака чӣ тавр ба функсия равон карда мешаванд?

Масъалаҳо барои кори мустақилона

Масъалаҳои гурӯҳи ишорагар ва суроға

1. Қимати ду тағйирёбандаи бутуни a ва b дода шудааст. Барои ин тағйирёбандаҳо ду ишорагар истифода бурда шавад. Тавассути ишорагар қимати тағйирёбандаи a -ро ду маротиба зиёд карда, сипас ҷойи қимати a ва b иваз карда шавад.
2. Қимати ду тағйирёбандаи бутуни a ва b дода шуда аст. Барои ин тағйирёбандаҳо ду ишорагар истифода бурда шавад. Тавассути ишорагарҳо қимати тағйирёбандаи a -ро ҳангоми иҷрои шари $a > b$ ду воҳид зиёд карда, дар ҳолати акс қимати b -ро се воҳид кам кунед.
3. Қимати ду тағйирёбандаи ҳақиқии a ва b дода шудааст. Барои ин тағйирёбандаҳо ду ишорагар истифода карда шавад. Тавассути ишорагар қимати тағйирёбандаи a -ро се маротиба зиёд карда, сипас тавассути ишорагарҳо ҷойи қимати онҳо иваз карда шавад.
4. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо дурустии шари $a < b < c$ санчида шавад.
5. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шари $a < b < c$ тавассути ишорагарҳо қимати онҳо ду воҳид зиёд карда шавад.
6. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шари $a < b < c$ тавассути ишорагарҳо қимати онҳо се воҳид кам карда шавад.
7. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шари $a > b > c$ тавассути ишорагарҳо қимати онҳо се воҳид кам карда шавад.
8. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой

- доштани шарти $a > b > c$ тавассути ишорагарҳо қимати онҳо ду воҳид зиёд карда шавад.
9. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шарти $a < b < c$ тавассути ишорагарҳо қимати онҳо ду воҳид зиёд карда шавад.
 10. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо қимати a – ро ду воҳид, қимати b – ро чор воҳид ва қимати c – ро се воҳид зиёд намоед.
 11. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо қимати a – ро панҷ воҳид, қимати b – ро ду воҳид ва қимати c – ро шаш воҳид зиёд намоед.
 12. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо қимати a – ро ду воҳид зиёд, қимати b – ро чор воҳид кам ва қимати c – ро ду маротиба зиёд намоед.
 13. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо қимати a – ро ду воҳид кам, қимати b – ро чор воҳид зиёд ва қимати c – ро ду маротиба кам намоед.
 14. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шарти $a < b < c$ тавассути ишорагарҳо қимати онҳо ду воҳид зиёд ва дар ҳолати акс 5 воҳид кам карда шавад.
 15. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шарти $a > b = c$ тавассути ишорагарҳо қимати онҳо ду воҳид зиёд ва дар ҳолати акс 5 воҳид кам карда шавад.
 16. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шарти $a = b < c$ тавассути ишорагарҳо қимати онҳо ба аломати баръакс гардонида шавад, дар ҳолати акс бошад, ду маротиба зиёд карда шавад.
 17. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо санчида шавад, ки оё онҳо тарафҳои секунҷаро ташкил медиҳанд ё не.
 18. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Тавассути ишорагарҳо санчида шавад, ки оё муодилаи квадрати $ax^2 + bx + c$ решаҳои ҳақиқӣ дорад не.
 19. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой

доштани шарти $a < b < c$ тавассути ишорагарҳо қимати онҳо ба қимати миёнаи арифметикиашон иваз карда шавад, дар ҳолати акс бошад, ду воҳид зиёд карда шаванд.

20. Қимати се тағйирёбандаи ҳақиқии a, b ва c дода шудааст. Барои ин тағйирёбандаҳо се ишорагар истифода карда шавад. Ҳангоми ҷой доштани шарти $a < b < c$ тавассути ишорагарҳо қимати онҳо ба қимати миёнаи геометрияшон иваз карда шавад, дар ҳолати акс бошад, ду воҳид зиёд карда шаванд.

Масъалаҳои гурӯҳи ишорагар ва ҳофизаи динамикӣ

1. Ду ишорагари типии бутун муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид карда, сипас қимати тағйирёбандаи якум 2 мартаба кам карда шавад.
2. Ду ишорагари типии бутун муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид карда, Сипас қимати тағйирёбандаи дуюм 3 мартаба зиёд карда шавад.
3. Се ишорагари типии рамзӣ (символӣ) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид кунед.
4. Ду ишорагари типии мантикӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда мувофиқан қиматҳои true (1) ва false (0) бахшед.
5. Ду ишорагари типии ҳақиқӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед.
6. Се ишорагари типии ҳақиқӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшида, сипас қимати ишорагари якум ду мартаба кам карда шавад.
7. Се ишорагари типии бутун муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшида, сипас қимати ишорагари якум ду мартаба зиёд ва қимати ду ишорагари баъдӣ 3 воҳид кам карда шавад.
8. Ду ишорагари типии бутун муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид карда, Сипас ҷойи қимати онҳоро иваз кунед.
9. Се ишорагари типии ҳақиқӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид карда, сипас ҷойи қимати дуто ишорагари авваларо иваз кунед.
10. Ду ишорагари типии мантикӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда мувофиқан

- қиматҳои true (1) ва false (0) бахшеда, сипас ҷойи қимати онҳо иваз карда шавад.
11. Ду ишорагари типии сатри муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Сипас барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид намоед.
 12. Як ишорагари типии бутун муаррифӣ карда, барои он ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаи ҷудокардашуда аз калавиатура қимат ворид карда, сипас онро 3 мартиба зиёд кунед.
 13. Се ишорагари типии рамзӣ (символӣ) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид кунед. Сипас ҷойи қимати ду ишорагари охиرونро иваз намоед.
 14. Ду ишорагари типии ҳақиқӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед. Қимати миёнаи арифметикии онҳоро ҳисоб кунед.
 15. Як ишорагари типии ҳақиқӣ муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед.
 16. Се ишорагари типии ҳақиқӣ (a, b, c) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед. Ҳангоми иҷроиши шарт $a < b < c$ қимати онҳо ду маротиба кам карда шавад.
 17. Се ишорагари типии ҳақиқӣ (a, b, c) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед. Ҳангоми иҷроиши шарт $a < b < c$ қимати ишорагари якум ду маротиба кам ва қимати ду ишорагари баъдӣ 3 воҳид зиёд карда шавад.
 18. Ду ишорагари типии бутун (a, b) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда аз калавиатура қимат ворид карда, ҳангоми ҷой доштани шарт $a < b$ ҷойи қимати онҳоро иваз кунед.
 19. Се ишорагари типии ҳақиқӣ (a, b, c) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед. Ҳангоми иҷроиши шарт $a > b > c$ қимати ишорагари якум ду маротиба кам ва қимати ду ишорагари баъдӣ 3 воҳид зиёд карда шавад.
 20. Се ишорагари типии ҳақиқӣ (a, b, c) муаррифӣ карда, барои онҳо ҳофизаи динамикӣ ҷудо намоед. Барои ҳофизаҳои ҷудокардашуда ба таври тасодуфӣ қимат бахшед. Ҳангоми иҷроиши шарт $a < b = c$ қимати ишорагари якум ду маротиба зиёд ва қимати ду ишорагари баъдӣ се маротиба кам карда шавад.

Масъалаҳои гурӯҳи массивӣ динамикӣ

1. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ рақами тартиби ва ҳосили зарби элементҳои аз 5 хурди онро ҳисоб кунед.
2. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ рақами тартиби ва ҳосили ҷамъи элементҳои аз $a(5)$ хурди онро ҳисоб кунед.
3. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ муайян кунед, ки элементҳои нахустини массиви a дар массиви b чанд маротиба во мекунад.
4. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ муайян кунед, ки элементҳои дуҷумлаи массиви a дар массиви b чанд маротиба во мекунад.
5. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ муайян кунед, ки элементҳои охири массиви a дар массиви b чанд маротиба во мекунад.
6. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ муайян кунед, ки элементҳои мусбати кадоме аз ин массивҳо бештар аст?
7. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ муайян кунед, ки элементҳои манфии кадоме аз ин массивҳо бештар аст?
8. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ элементҳои мусбати он ба квадрат бардошта шаванд.
9. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ элементҳои манфии он ду воҳид кам карда шаванд.
10. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ муайян кунед, ки элементҳои мусбати кадоме аз ин массивҳо камтар аст?
11. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ва $b(m)$ элементҳои мусбати массиви a — ро панҷ воҳид зиёд ва элементҳои манфии массиви b — ро ду воҳид кам кунед.
12. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ элементҳои мусбати онро ба 25 иваз кунед.
13. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки аз элементҳои охири хурданд ду воҳид зиёд карда, элементҳои боқимондари 5 воҳид кам кунед.
14. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки аз адади додашудаи s хурданд ду воҳид зиёд карда, элементҳои боқимондари 2 маротиба кам кунед.
15. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки аз адади додашудаи s

калонанд се воҳид зиёд карда, элементҳои боқимондари 2 маротиба кам кунед.

16. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки шарт $a(0) < a(i) < a(n-1)$ –ро қонеъ мекунонд, се воҳид зиёд карда, элементҳои боқимондари 2 воҳид кам кунед.
17. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ суммаи ҳамаи элементҳоеро, ки шарт $a(0) < a(i) < a(n-1)$ –ро қонеъ мекунонд ҳисоб кунед.
18. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки аз элементи дуюм калонанд 5 воҳид кам кунед.
19. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки аз решаи квадратии адади додашудаи d хурданд се воҳид зиёд карда, элементҳои боқимондари ду маротиба кам кунед.
20. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳоеро, ки аз адади додашудаи d калонанд ба квадрат бардошта, элементҳои боқимондари ду маротиба зиёд кунед.

Масъалаҳои гурӯҳи ишорагарҳо ва мураттабсозӣ

1. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои мусбати онро ба 3 зарб карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мураттабсозии *tag* онҳоро мураттаб созед.
2. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии онро ба 3 зарб карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мураттабсозии хубобӣ онҳоро мураттаб созед.
3. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои мусбати онро ба 3 зарб карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мураттабсозии интихобӣ онҳоро мураттаб созед.
4. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои мусбати онро ба 3 тақсим карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мураттабсозии зуд (*quick*) онҳоро мураттаб созед.
5. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии онро ба 3 тақсим карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мураттабсозии *tag* онҳоро мураттаб созед.
6. Бо истифода аз ишорагарҳои массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои мусбати онро дар массиви $c(m)$

- ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии гузориши онҳоро мурааттаб созед.
7. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои мусбати онро дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии гузориши онҳоро мурааттаб созед.
 8. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии онро дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи зуд онҳоро мурааттаб созед.
 9. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии онро дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии зуд онҳоро бо тартиби камшавӣ мурааттаб намоед.
 10. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии онро ба квадрат бардошта, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии интиҳоб онҳоро бо тартиби камшавӣ мурааттаб намоед.
 11. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ решаи квадрати ҳамаи элементҳои мусбати онро дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас бо истифода аз методи мурааттабсозии зуд онҳоро бо тартиби камшавӣ мурааттаб кунед.
 12. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии ду воҳид кам карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии хубобӣ онҳоро мурааттаб созед.
 13. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ квадрати ҳамаи элементҳои мусбати онро ба 3 зарб карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии интиҳобӣ онҳоро мурааттаб созед.
 14. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои ҷуфти онро, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии quick) онҳоро мурааттаб созед.
 15. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои ҷуфти манфии онро ба -2 зарб карда, дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии tag онҳоро мурааттаб созед.
 16. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои тоқи онро дар массиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии гузориши онҳоро мурааттаб созед.
 17. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои шарти $a(0) < a(i) < a(n -$

- 1) – ро қаноаткунонандаи онро дар масиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи мурааттабсозии гузориши онҳоро мурааттаб созед.
18. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои шартӣ $a(i) < 35$ – ро қаноаткунонандаи онро дар масиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи зуд онҳоро мурааттаб созед.
19. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои манфии онро ба аломати баръакс иваз карда, дар масиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи зуд онҳоро бо тартиби камшавӣ мурааттаб намоед.
20. Бо истифода аз ишорагарҳо массиви динамикӣ муаррифӣ намоед. Бо дода шудани массиви $a(n)$ ҳамаи элементҳои соддаи онро дар масиви $c(m)$ ($m \leq n$) нигоҳ доред (бахшед). Сипас, тавасути методи интиҳоб онҳоро бо тартиби камшавӣ мурааттаб намоед.

Боби 8. Кор бо сатрҳо ва рамзҳо

Муқаддима

Ҳар гуна матн дар компютер ба намуди ададӣ нигоҳ дошта мешавад. Мувофиқи стандарти кодгузори ASCII (American standard code for Information Interchange) ҳар рамз дорои коди махсусе мебошад. Дар забони с++ додаҳои навъи char низ ҳамчун адади бутун фаҳмида шуда, фосилаи қиматҳои аз 0 то 255 мебошад. Наҳви муаррификунии додаҳои навъи мазкур намуди зеринро доранд:

```
char *str="Salom!"
```

Барои сатри мазкур, дар забони С++ ҳафт байт ҳофиза ҷудо карда мешавад. Ҳар як рамз як байт фаҳмида шуда, нишонаи ба охир-расии сатр низ ҳамчун як байт фаҳмида мешавад.

83	97	108	111	109	33	0
'S'	'a'	'l'	'o'	'm'	'\0'	(null)

Дар компютер як байт аз 8 бит иборат мебошад, дар навбати худ ҳар бит метавонад аз ду қимати системаи ҳисоби дуи 0 ё 1 -ро қабул кунад. Аз ин рӯ, сатри мо намуди зеринро мегирад.

Ҳарф	S	a	L	O	m	!
ASCII-коди ҳарф	83	97	108	111	109	33
Битҳои код	1110011	1110000	1110110	1110111	1110110	0110000

Ба тағйирёбандаҳои рамзӣ бахшидани қимати аввала

Тағйирёбандаҳои рамзӣ низ монанди тағйирёбандаҳои дигар муаррифӣ карда мешаванд ва ҳангоми муаррификунии чунин тағйирёбандаҳо метавон ба онҳо қимати аввала бахшид.

Дар забони С++ сатрҳо ҳамчун массивҳои рамзӣ шинохта шуда, охири он ба рамзи Null (охири сатр) ба поён мерасад. Дар дохили барнома рамзи мазкур дар қолаб (форма)-и escape square бо '\0' нишон дода мешавад. Зернависи поёнии элементҳои массиви навъи сатрҳо 0 буда, зернависи болоӣ бошад, якто кам аз миқдори рамзҳои ҳамон сатр ба ҳисоб меравад. Масалан, сатри «Hello world» як массиви рамзии дорои 11 рамз мебошад. Дар ин навъ массивҳо ҷойи ҳолӣ низ ҳамчун як элемент фаҳмида мешавад. Ба массивҳои сатрӣ дар вақти муаррифӣ намудан, метавон қимати аввала бахшид. Ба дастури зерин диққат диҳед:

```
char Arsatr[]="Hello World";
```

Массиви муаррифишуда дорои 11 элемент мебошад. Инчунин массивҳои сатрӣ метавонанд ҳар як элементашон алоҳида бошад. Масалан, массиви Arsatr-ро метавон ба сурати зерин низ эълон кард:

```
char Arsatr[]={ 'H', 'e', 'l', 'l', 'o', '\0', 'W', 'o', 'r', 'l', 'd' }
```

Тавре ки дида мешавад, дар ин ҷо ҳар як элементи массиви рамзӣ дар дохили ноҳунаки якто-якто (апостроф) навишта шуда, барои ҷойи ҳолӣ (фосила) рамзи '\0' истифода шудааст ва оператори коммо бошад, барои аз ҳамдигар ҷудо намудани рамзҳо ба кор рафтааст. Барои дар ин навъ массивҳо

ба элементи зарури дастрас шудан аз зернависи (рақами тартиби)-и рамзҳо истифода бурда мешавад.

Монанди дигар массивҳо, зернависи элементҳои массиви рамзӣ низ аз нол оғоз ёфта, ба n-1 ба охир мерасад, яъне дар массиви мазкур элементи нолум ин рамзи (харфи) «Н» буда, элементи n-1 (10)-ум рамзи «d» мебошад.

Мисол. Барномае нависед, ки ҳамаи элементҳои массиви `Arsatr[]`-ро ба чоп диҳад. Инчунин элементи аввалин ва охиринашро низ алоҳида ба чоп диҳад.

```
#include <iostream>
using namespace std;
int main(){
    const int n=11;
    cout<<"Massivi dodoshuda=";
    char Arsatr[]={'H','e','l','l','o','\0','W','o','r','l','d'};
    for (int i=0;i<n;i++)
        cout<<Arsatr[i];
    return 0;
}
```

Чи хеле ки мебинем, дар ин ҷо аввал доимие муаррифӣ шуда аст, пас аз он массив бо элементҳои аввалааш таъриф шудааст. Оператори даврӣ то замони ба охир расидани элементҳои массив такрор шуда, дар ҳар итератсия (давр ё кадам) як элементро ба саҳифаи тасвир хориҷ мекунад.

Ҳангоми сатрҳоро ба намуди муқаррарӣ эълон кардан, метавон қимати онҳоро аз клавиатура ворид кард. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    char chr[100];
    cout<<"Enter you name, please "<<endl;
    cin>>chr;
    cout<<"You name is:"<<chr<<endl;
    return 0;
}
```

Натиҷаи барнома шакли зеринро дорад:

```
Enter you name, please
Ali
You name is: Ali
```

Чунончи аз мисол бармеояд, ҳангоми ворид кардани сатр оператори дохилкунии `cin` то мавқеи фосила рамзҳоро қабул кардаасту ҳалос. Рамзҳои пас аз ҷои ҳолӣ омада хонда нашуданд. Барои дастрас шудани тамоми сатр аз функсияи `gets` истифода бурда мешавад. Аз ин ҷо барномаи боло намуди зеринро мегирад.

```
#include <iostream>
using namespace std;
int main(){
    char chr[100];
    cout<<"Enter you name, please "<<endl;
```

```

    gets(chr);
    cout<<"You name is:"<<chr<<endl;
    return 0;
}

```

Тавре ки дида мешавад, дар ин ҷо функсияи gets ба миқдори андозаи массив символҳои дохилкардашударо меҳонад.

Дар забони C++ тағйирёбандаҳои навъи char ҳамчун адади бутун фаҳмида мешавад. Аз рӯи кодгузори ASCII ҳар як рамз дорои қи-мати ададӣ аст. Аз ин хотир, метавон элементҳои массивро рамз ба рамз дохил кард. Мисол:

```

#include <iostream>
using namespace std;
int main(){
    char chr[30];
    int n=26;
    char s='a';
    for (int i=0;i<n;i++)
        chr[i]=s+i;
    chr[n]='\0';
    cout<<chr<<endl;
    return 0;
}

```

Натиҷаи барнома ба сурати зерин аст.

abcdefghijklmnopqrstuvwxyz

Функсияҳои кор бо рамзҳо

Ҳамаи маълумот ба компютер ҳамчун рамзҳо дохил карда мешаванд. Рамз ягон ҳарф ишора ва ё рақам мебошад. Сатр бошад, аз пайдарпайии рамзҳо таркиб ёфтааст. Дар забони C++ барои кор кардан ба рамзҳо якчанд функсияҳо мавҷуданд, ки ҳангоми пайваст намудани китобхонаи <ctype.h> барои истифодабаранда дастрас мегарданд. Дар ҷадвали зерин рӯйхати якчандто аз ин функсияҳо оварда шудааст.

№	Номи функсия	Маъно ва мафҳум
1	int isdigit (int c)	Дорои қимати true аст, агар аргументи он яке аз рақамҳои аз 0 то 9 бошад, дар акси ҳол дорои қимати false мегардад.
2	int isalpha(int c)	Дорои қимати true аст, агар аргументи он ҳарф бошад, дар акси ҳол дорои қимати false мегардад.
3	int isalnum(int c)	Дорои қимати ҳақ аст, агар аргументи он ҳарф ва ё рақам бошад.
4	int isxdigit (int c)	Дорои қимати true аст, агар аргументи он адади системаи шонздаҳӣ бошад, дар акси ҳол дорои қимати false мегардад.
5	int islower(int c)	Дорои қимати true аст, агар аргументи он ҳарфи хурд бошад, дар акси ҳол дорои қимати false мегардад. Функсия танҳо барои ҳарфҳои латинӣ кор мекунад.

6	int isupper (int c)	баръакси функцияи islower(int c) амал мекунад.
7	int tolower (int c)	Агар аргументи ин функция ҳарфи калон бошад, онро ба ҳарфи кучак табдил медиҳад .
8	int toupper (int c)	Баръакси функцияи tolower (int c) мебошад.
9	int isspace(int c)	ба қимати true баробар аст, агар аргументи он яке аз рамзҳои ҷудокунанда бошанд ба монанди: гузаштан ба сатри нав('\n'), гузаштан ба саҳифаи нав ('\f'), гузаштани сархати уфуқӣ ('\t'), гузаштани сархати амудӣ ('\v'), бозгашти навард (carriage return) ('\r') дар акси ҳол бошад, дорои қимати false мешавад.
10	int ispunct(int c)	ба қимати ҳақ доро аст, агар аргументи он рамзи пунктуатсия бошад. Рамзҳои пун-туаксия: фосила, рақам, ҳарф ва ғайра мебошанд.
11	int isprint(int c)	Ба қимати ҳақ доро аст, агар аргументи он симболи чопшаванда бошад.
12	isgraph(int c)	Ба қимати ҳақ доро аст, агар аргументи он сим-воли чопи бошад, ба ғайр аз фосила.

Қайд. Бозгашти навард (carriage return)- симболи идоракунӣ механизмест, ки барои мавқеи курсорро ба аввали сатр овардан хизмат мекунад.

Мисол. Истифодаи функцияҳои isdigit(), isalpha(), isalnum() ва isxdigit(). Барномае нависед, ки бо дохил кардани рамз муайян кунад, ки он рақам, ҳарф, ва ё рақами системаи ҳисоби шонздаҳӣ мебошад.

```
#include <iostream>
using namespace std;
int main(){
    char chr;
    int i=0;
    while(++i<4)
    {
        cout<<"chr=";<cin>>chr;
        cout<<"Simvoli "<<chr<<(isdigit(chr)? " raqam hast": "
raqam nest")<<endl;
        cout<<"Simvoli "<<chr<<(isalpha(chr)? " harf hast": " harf
nest")<<endl;
        cout<<"Simvoli "<<chr<<(isxdigit(chr)? " raqami sistemai
hisobi shonzdahi ast": " raqami sistemai hisobi shonzdahi
nest")<<endl;
        cout<<"Simvoli "<<chr<<(isalnum(chr)? " harf yo raqam
hast": " harf yo raqam nest")<<endl;
    }
    return 0;
}
```

Мисол. Истифодаи функцияҳои islower() ва isupper(). Барномае нависед, ки бо дохил кардани ҳарф муайян кунад, ки он хурд аст ё калон.

```
#include <iostream>
#include <ctype.h>
```

```

using namespace std;
int main(){
    char chr;
    int i=0;
    while(++i<3)
    {
        cout<<"chr=";<<cin>>chr;
        cout<<"Simvoli "<<chr<<(islower(chr)? " harfi hurd ast" : "
harfi hurd nest")<<endl;
        cout<<"Simvoli "<<chr<<(isupper(chr)? " harfi kalon ast": "
harfi kalon nest")<<endl;
    }
    return 0;
}

```

Функсияи get ва getline

Функсияи get барои дохилкунии як рамз истифода шуда, метавонад ҳангоми дохилкунии дорои аргумент ва ё беаргумент бошад. яъне, он метавонад, қимати воридшударо ба аргументи худ ва ё ба ягон тағйирёбандаи дигар бахшад. **Мисол:**

```

#include <iostream>
#include <ctype.h>
using namespace std;
int main(){
    char chr;
    chr=cin.get();
    cout.put(chr);
    return 0;
}

```

Тавре ки дар ин мисол дида мешавад, функсияи мазкур якҷоя бо оператори дохилкунии cin истифода шуда, қимати воридшударо ба тағйирёбандаи chr бахшидааст. Илова бар ин, онро ба сурати зерин низ метавон истифода кард.

```

#include <iostream>
using namespace std;
int main(){
    char chr;
    cin.get(chr);
    cout.put(chr);
    cout.put(chr);
    return 0;
}

```

Аз ин мисол аён мегардад, ки функсияи get() тағйирёбандаи лозимиро ҳамчун аргумент қабул намуда, қимати онро вориди барнома намудааст. Ҳангоми дохил намудан якчанд рамз функсияи get() танҳо рамзи нахустинро мехонаду халос. Масалан, ҳангоми ворид кардани якчанд рамз натиҷа намуди зеринро мегирад.

Input: Computer

Output: C

Раши дуими истифодабарии ин функция ба аргументи рамзи то расидан ба нишонаи интиҳои матн пайхам рамзхоро мебахшад. Ин раширо баъдтар хангоми истифодабарии операторҳои идоракуни дида мебароем.

Раши сеюми истифодаи ин функция дорои се параметр мебошад. Массиви рамзи (ягон сатр), адади максималии миқдори рамзҳои массив ва рамзи идоракунанда хусусан ('\n'). Ин раши истифодабарии рамзхоро то замоне ки аз миқдори максималии дода-шуда як воҳид хурданд меҳонад, яъне (max-1)-рамзро меҳонад. Баъдан барои ба охирагии сатрҳои дохилшуда дар массиви рамзи ба сифати буфери барнома рамзи нули истифода бурда мешавад. Мисол:

```
#include <iostream>
using namespace std;
int main(){
    const int Size=5;
    char Satr[Size];
    cin.get(Satr,Size);
    cout<<Satr<<" \n ";
    return 0;
}
```

Натиҷаи барнома ба намуди зерин аст:

Computer
Comp

Функцияи `getline()` бошад, монанди раши сеюми функцияи `get()` буда, чи қадаре ки миқдори максималии бошад, ҳамон қадар рамзро хонда, рамзи маҳдудкунандаро аз ҷараён нест мекунад, яъне онро ба рамзи массив сабт намекунад. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    const int Size=5;
    char Satr[Size];
    cin.getline(Satr,Size);
    cout<<Satr<<" \n ";
    return 0;
}
```

Функцияи `мазкур`, бештар хангоми ягон файли маълумотро хондан, истифода бурда мешавад, ки дар оянда дар мавзӯи файлҳо муфассал ин матолибро мавриди баҳс қарор медиҳем.

Функцияҳои табдилдиҳии сатрҳо

Бисёр додаҳо дар аввал ба намуди матнӣ дода мешаванд, масалан, бигзор якчанд адад дар дохили як матн дода шуда бошанд, барои онҳоро истифода бурдан, ба мо лозим меояд, ки навъи онҳоро аз сатрӣ ба ададӣ гардонем. Барои иҷро ин амал, C++ ба мо имконият медиҳад, то аз функцияҳои, ки дар китобхонаи `<stdlib>` мавҷуданд, истифода барем. Дар ҷадвали поён якчандто аз

ин функцияҳо оварда шудааст. Барои бо ин функцияҳо кор кардан, бояд китобхонаи <stdlib> пайваст карда шавад.

№	Нахви функция	Маъно ва мафҳум
1	double atof (const char *nPtr)	Сатри nPt-ро ба адади навъи double табдил медиҳад.
2	double atoi (const char *nPtr)	Сатри nPt-ро ба адади навъи int табдил медиҳад.
3	double strtod (const char *nPtr)	Сатри nPt-ро ба адади навъи long int табдил медиҳад.
4	long strtol (const char *nPtr, char *endPtr, int base)	Сатри nPt-ро ба адади навъи long табдил медиҳад.
5	unsigned long strtoul (const char *nPtr, char *endPtr, int base)	Сатри nPt-ро ба адади навъи unsigned long табдил медиҳад.

Истифодаи функцияҳои табдилдиҳии сатрҳо

1) Функцияи atof(), atoi() ва atol() барои додаҳои ададии дар дохили ноҳунак навишта ва ё дар файл ҷойгирбударо мувофиқан ба намудҳои ҳақиқӣ ва бутуни дароз (long) табдил додан хизмат мекунад. **Мисол:**

```
#include <iostream>
#include <stdlib.h>
using namespace std;
int main(){
    double Db;
    int In;
    long Lg;
    char *d,*I,*L;
    d="9.25";
    I="254";
    L="1245748";
    Db=(atof(d)+21)/4;
    In=25*atoi(I)-65;
    Lg=254-atol(L);
    cout<<"Db="<<Db<<"\nIn="<<In<<"\nLg="<<Lg<<endl;
    return 0;
}
```

Тавре ки дида мешавад, дар аввал тағйирёбандаҳои d, I ва L аз навъи char муаррифӣ гардидаанд ва баъд ба онҳо қимате ихтисос дода шудааст. Сипас, дар ифодаҳои муҳосиботӣ бо истифодаи аз функцияҳои atof(), atoi() ва atol() навъи онҳо ба намуди ададӣ гардонидани шудааст. Баъдан бо ёрии оператори хоричкуни қимати ифодаҳои муҳосиботӣ ба ҷоп дода шудааст.

2) Функцияи strtod() - функция дорои ду аргумент, яке аз навъи char ва дигаре аз навъи ишорагар буда, қимати ададиро аз дохили сатр ҷудо мекунад (бармегардонад). Тағйирёбандаи ишорагар бошад, қисми боқимондаи сатрро бармегардонад, ин дар ҳолатест, ки агар адад дар аввали сатр бошад. Мисол:

```
#include <iostream>
using namespace std;
int main(){
```



```

double d;
const char *chr="25.5 adadi haqiqi ast";
char *chr1;
d=strtod(chr,&chr1);
cout<<"chr="<<chr<<"\nd="<<d<<"\nchr1="<<chr1<<endl;
return 0;
}

```

3) Функция `strtod()` аз пайдапайии рамзҳо адади навъи `long`-ро бар мегардонад. Функцияи мазкур дорои се параметр аст, ки якеаш сатри додасуда, дигараш ишорагар ба сатр ва аргументи сеюм бошад, адади бутуни фосилаи 0 то 36 мебошад. Аксаран ба ҷойи аргументи сеюм адади нолро истифода мекунанд. Тағйирёбандаи ишорагар ҳамон қисматеро, ки пас аз гирифтани адади `long` боқӣ гузошта шудааст, дар худ нигоҳ медорад. Мисол:

```

#include <iostream>
using namespace std;
int main(){
    long d;
    const char *chr="-123456abc";
    char *chr1;
    d=strtod(chr,&chr1,0);
    cout<<"chr="<<chr<<"\nd="<<d<<"\nchr1="<<chr1<<endl;
    return 0;
}

```

4) Функцияи `strtoul()`-монанди функцияи `strtod()` аз пайдарпайии символҳо адади дарози беаломат (`unsigned`)-ро бармегардонад. Функцияи мазкур низ дорои се параметр мебошад, ки параметри якум, сатре ки бояд онро ба адад табдил диҳем, параметри дуввум тағйирёбандаи ишорагари сатрӣ, ки бақияи сатри додасударо бармегардонад ва параметри саввум адади бутуни аз 0 то 36 мебошад. **Мисол:**

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    long d;
    const char *chr="246810abc1245";
    char *chr1;
    d=strtoul(chr,&chr1,0);
    cout<<"chr="<<chr<<"\nd="<<d<<"\nchr1="<<chr1<<endl;
    return 0;
}

```

Функцияҳои ҷустуҷӯи аз китобхонаи кор бо сатрҳо

Ҳангоми навиштани барномаҳои калонтар, дар онҳо кор кардан ба сатрҳо ва рамзҳо лозим аст. Баъзан лозим мешавад, як сатрро аз дохили сатри дигар ҷустуҷӯ кунем. Барои иҷрои ин амалҳо C++ ба мо имконият медиҳад, ки аз функцияҳои тайёр, ки дар китобхонаҳои `<string.h>` ҷойгиранд, истифода намоем. Дар ҷадвали зерин якчандто аз он функцияҳо оварда шудаанд.

№	Наҳви функция	Маъно ва мафҳум
1	char *strchr(const char *S,int n)	Маҳалли аввалини рамзи с-ро дар сатри S ҷустуҷӯ мекунад, агар пайдо шуд, як ишорагар ба с баргардонида мешавад ва дар акси ҳол як ишорагари Null баргардонида мешавад.
2	char *strrchr(const char *S,int c)	Маҳалли охирини рамзи с-ро дар сатри S ҷустуҷӯ мекунад, агар пайдо шуд, як ишорагар ба с баргардонида мешавад ва дар акси ҳол як ишорагари Null баргардонида мешавад.
3	size_t strspn (const char *S,const *chsr S2)	Қисми аввалаи сатри S1, ки танҳо аз рамзҳои сатри S2 иборат аст муайян мекунад.
4	char *strpbrk(const char *S,const *chsr S2)	Маҳалли вуқуъи аввалини символи сатри S2-ро дар сатри S1 ҷустуҷӯ мекунад, агар пайдо шуд, як ишорагар ба ин рамз дар S1 баргардонида мешавад. Дар ҳолати акс қимати '\0' бармегардонад.
5	size_t strcspn(const char *S,const *chsr S2)	Қисми аввалаи сатри S1, ки дар он рамзҳои сатри S2 намехобанд, муайян мекунад.
6	char *strstr(const char *S,const *chsr S2)	Маҳалли аввалини ҷойгиршавии сатри S1-ро дар сатри S1 муайян мекунад, агар пайдо шавад, як ишорагар дар дохили S1 баргардонида мешавад ва дар ҳолати акс як ишорагари Null баргардонида мешавад.

Тавре ки аз ин чадвал дида мешавад, функцияҳои strcspn() ва strspn() дорои навъ (type)-и size_t мебошанд. Ин навъи додасуда дар ҳолати стандарти ҳамчун навъи бутун буда, амали sizeof-ро муайян мекунад, яъне он ба навъҳои unsigned int ё unsigned long муродиф мебошад.

Мисол. Истифодаи функцияи strchr().

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    const char *S="matematica";
    char c='e';
    cout<<strchr(S,c)<<endl;
    return 0;
}
```

Натиҷаи барнома чунин аст: ematica

Истифодаи функцияи strcspn()-ин функция сатри якум ва дуумро бо ҳам муқоиса мекунад ва миқдори рамзҳои сатри якум ҳастанду дар дуум нестанд, бармегардонад.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    const char *S1="Qimati Pi=3.14";
```

```
char *S2="1234567890";
cout<<strcspn(S1,S2)<<endl;
return 0;
}
```

Натиҷаи барнома чунин аст: 10

Мисол. Истифодаи функсияи `strpbrk()`-ин функсия ду сатрро муқоиса мекунад. Агар ягон рамзи сатри дуюм дар сатри якум мавҷуд бошад, ишорагар онро бармегардонад, вагарна қимати `null` бармегардонад.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    const char *S1="Qimati Pi=3.14 meboshad";
    char *S2="Adadi doimi ast.";
    cout<<strpbrk(S2,S1)<<endl;
    return 0;
}
```

Ин барнома ҳар ду сатрҳоро муқоиса карда, шурӯъ аз он рамзе, ки бо он сатри якум ба охир расидааст, сатри дуввумро бармегардонад.

Функсияҳои кор бо ҳофиза аз китобхонаи коркарди сатрҳо

Китобхонаи коркарди сатрҳо дорои функсияҳои зиёде ҳастанд, ки барои нусхабардорӣ, муқоисакунӣ ва ҷустуҷӯи қисмате аз хотира истифода мешаванд. Функсияҳои коркарди сатрҳо ба сифати ба охирирасӣ ё анҷомёбӣ қимати `'\0'` ё `Null` –ро қабул мекунанд, зеро ин функсия амали массиви рамзиро иҷро мекунад.

№	Наҳви функсия	Маъно ва мафҳум
1	<code>void *memcpy(void *S1, const void *S2, size_t n)</code>	n-рамзи нишондодашударо аз объекти S2 нусха-бардорӣ карда, ба S1 мегузаронад.
2	<code>void *memmove(void *S1, const void *S2, size_t n)</code>	n-рамзи объектеро, ки дар сатри S2 нишон додааст, нусхабардорӣ карда, ба сатри S1 мегузаронад.
3	<code>int *memcmp(const void *S1, const void *S2, size_t n)</code>	n-рамзи нишондодашудаи объекти S1-ро ба S2 муқоиса мекунад. Агар онҳо бо ҳам баробар бошанд, қимати 0, ҳангоми S1 аз S2 хурд будан, қимати манфӣ ва дар акси ҳол, қимати мусбат бармегардонад.
4	<code>void *memchr(const void *S, int c, size_t n)</code>	Рамзи аввалаи дохилкардашударо муайян намуда, ба <code>unsigned char</code> табдил медиҳад ва мавқеи аввалаи аргументи c-ро дар объекти S ҷустуҷӯ мекунад, агар пайдо шуд, ишорагар онро бармегардонад. Дар акси ҳол қимати <code>Null</code> бармегардонад.
5	<code>void *memset(void *S, int c, size_t n)</code>	Барои буферро ба рамзи дар параметри c нишон-додашуда пур

	кардан истифода мешавад. Дар ин ҷо *S ишорагар ба буфер буда, n- микдори рамзхое, ки бояд ба буфер пур карда шаванд, мебошад.
--	---

1) Истифодаи функсияи memcpy().

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char S1[10];
    char S2[]="Copy the string";
    memcpy(S1,S2,10);
    cout<<S1<<endl;
    return 0;
}
```

Ин барнома 10 рамзи сатри S2-ро ба сатри S1 нусхабардорӣ мекунад. Натиҷаи барнома чунин аст: Copy the s

2) Истифодаи функсияи memmove().

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char S1[10];
    char S2[]="Copy the string";
    memmove(S1,S2,10);
    cout<<S1<<endl;
    return 0;
}
```

Натиҷаи ин барнома низ монанди мисоли 1 аст.

3) Истифодаи функсияи memcpy().

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char S1[]="Students";
    char S2[]="books";
    memcpy(S1,S2,4);
    cout<<S1<<endl;
    return 0;
}
```

Ин барнома 4 рамзи сатри S2 – ро ба сатри S1 нусхабардорӣ мекунад. Пас аз нусхабардорӣ чор рамзи аввали сатри S1 нест мешаванд. Натиҷаи барнома чунин намуд дорад: bookents

4) Истифодаи функсияи memchr().

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char S[]="It is string"; char S1='r';
```

```

cout<<static_cast<char *>(memchr(S,S1,20));
return 0;
}

```

Ин барномаи саркарда аз рамзи S1- рамзҳои сатри S-ро ба чоп медиҳад. Дар ин ҷо static_cast - оператори табдилдиҳӣ буда, натиҷаи функсияи memchr()-ро ба намуди рамзӣ табдил медиҳад. Дар сурати сарфи назар кардани он ASCII коди рамзҳо ба чоп бароварда мешаванд. Натиҷаи барнома чунин аст: restrstring

5) Истифодаи функсияи memset()

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    char S[]="AAAAAAAAAA";
    char S1='s';
    cout<<static_cast<char *>(memset(S,S1,5))<<endl;
    return 0;
}

```

Ин барнома ба ҷои 5 рамзи аввали сатри S 5 то рамзи дар параметри S1 нишондодашударо мегузорад. Натиҷаи барнома чунин аст: sssssAAAAA

Функсияҳои кор бо сатрҳо ва рамзҳо

Барои кор кардан ба сатрҳои матнӣ (риштаҳо) дар забони C++ онҳоро ҳамчун массивҳои рамзӣ муаррифӣ менамоянд. Ҷои холигӣ низ ҳамчун як рамз фаҳмида мешавад. Дар ҳар забони барноманависӣ барои кор кардан ба сатрҳо функсияҳои стандартӣ мавҷуданд. Дар забони C++ низ функсияҳои зиёде мавҷуданд, ки дар ҷадвали зерин муҳимтарин аз онҳо гирд оварда шудаанд.

№	Функсия	сарлавҳа	Маъно ва мафҳум
1	strcpy(s1,s2)	<string.h>	Сатри s2-ро ба сатри s1 нусхабардорӣ мекунад.
2	strcat (s1,s2)	<string.h>	Сатри s2-ро аз қафои сатри s1 пайваст мекунад.
3	strcmp (s1,s2)	<string.h>	Сатри s2-ро ба сатри s1 муқоиса мекунад, агар онҳо баробар бо-шанд, қимати 0 бармегардонад.
4	strpbrc(const char S1,const char S2)	<string.h>	Сатрҳои S1 ва S2 ро ба ҳам муқоиса мекунад. Агар ягон рамзи S2 дар S1 мавҷуд бошад, ишорагар онро бармегардонад
5	strset(char_c, int_ch)	<string.h>	Барои ба ҷойи сатри додашуда ба миқдори рамзҳои чоп намудани адад хизмат мекунад.
6	strchr(s,ch)	<string.h>	Рамзи охирини воридшударо ба рамзҳои сатр муқоиса мекунад, агар пайдо шуд, ишорагар онро бармегардонад.
7	strstr(s1,s2)	<string.h>	Аз рамзҳои сатри S2 сар карда сатри S1-ро ба чоп медиҳад.

8	strupr(*_s)	<string.h>	Рамзҳои сатри *_s ба registerи болоӣ табдил медиҳад
9	strlen(*s)	<string.h>	Дарозии сатри s-ро муайян мекунад.
10	strrev(*S)	<string.h>	Сатри S-ро баракс мегардонад. (Масалан *S=roh; strrev(S)=hor)
11	strlwr(*S)	<string.h>	Баръакси функсияиstrupr() амал мекунад.
12	tolower(ch)	<type.h>	Рамзи додашударо аз ҳарфи хурд ба ҳарфи калон табдил медиҳад.
13	toupper(ch)	<type.h>	Рамзи додашударо аз ҳарфи калон ба ҳарфи хурд табдил медиҳад.

Функсияҳои кор бо сатрҳо ва рамзҳо дар китобхонаҳои <string.h>, <stdlib.h>, ва <type.h> мавҷуданд. Функсия дар ҳамон ҳолат ба кор медарояд, ки агар китобхонаи вобастаи ба он пайваст карда шудааст.

Мисол. Барномае нависед, ки дар он функсияҳои чадвали мазкур истифода шавад.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char chr[30],*b="Hello ",*c="World";
    cout<<"Istifodai az function-i
    strcpy\n" ;
    strcpy(chr,b);
    cout<<chr<<endl;
    cout<<"Istifodai function-i strcat\n" ;
    strcat(b,c);
    cout<<b<<endl;
    cout<<"Istifodai function-i strcmp\n" ;
    cout<<strcmp(b,c)<<endl;
    return 0;
}
```

Дар ин мисол функсияи strcpy қимати тағйирёбандаи рамзии b-ро ба тағйирёбандаи рамзии chr нусхабардорӣ мекунад. Функсияи strcat бошад, қимати тағйирёбандаи рамзии c-ро дар қафои тағйирёбандаи рамзии b пайваст мекунад ва охири функсияи strcmp дар оператори хориҷкунӣ қимати сатри b ва c-ро муқоиса мекунад. Аз сабаби он ки сатрҳои мазкур баробар нестанд, функсия қимати манфӣ бармегардонад. Натиҷаи барнома шакли зеринро дорад:

```
Istifodai function-i
strcpy
Hello
Istifodai function-i strcat
Hello World
Istifodai function-i
strcmp
-39
```

Истифодаи функсияи strcpy() ва strcat ()

Тавре ки гуфта будем, функсияи strcpy() сатри s2-ро ба сатри s1 нусхабардорӣ мекунад. Функсияи strcat () бошад, сатри s2-ро аз қафои сатри s1 пайваст мекунад. Ин ҳарду функсияро дар мисоли зерин дида мебароем.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char str[400];
    char namefirstName[100];
    char sur[100];
    char kor[100] ;
    cout<<"Nomatonro vorid namuda tugmachai Enter-ro zer kuned\n";
    cin.getline(namefirstName,90);
    cout<<"Addressatonro vorid namuda tugmachai Enter-ro zer
kuned\n";
    cin.getline(sur,90);
    cout<<"Joi koratonro vorid namuda tugmachai Enter-ro zer
kuned\n";
    cin.getline(kor,90);
    strcpy(str,"\n Nom LastName: " );
    strcat (str,namefirstName);
    strcat (str," \nAddress: " );
    strcat (str,sur);
    strcat (str," \nzoyi kor: ");
    strcat (str,kor);
    cout<<str<<endl;
    return 0;
}
```

Истифодаи функсияи strset(char_c, int_ch)

Функсияи strset(char_c, int_ch) бошад, барои ба ҷойи сатри до-дашуда ба миқдори рамзхояш ҷоп намудани адад хизмат мекунад, ҳоло иҷроиши ин функсияро бо як мисоли кӯчак дида мебароем.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    char string[10] = "123456789";
    char symbol = '1';
    printf("Pesh az strset(): %s\n", string);
    strset(string, symbol);
    printf("Pas az strset(): %s\n", string);
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

Pesh az strset():123456789

Pas az strset(): 111111111

Массивҳои дученака аз навъи рамзӣ

Массивҳои якченака аслан барои як сатри рамзӣ истифода бурда мешаванд. Барои якчанд сатрро муаррифӣ ва истифода кардан аз массивҳои дученакаи навъи рамзӣ истифода намудан лозим меояд. Дар ин ҳолат зернависи якуми массив миқдори сатрҳо ва зернависи дуввум бошад, дарозии сатрҳоро муайян мекунад. **Мисол:**

```
#include <iostream>
using namespace std;
int main(){
    char s[5][10]={"Toirov","Dodarov","Aliev","Saidov","Valiev"};
    for (int i=0;i<5;i++)
        cout<<s[i]<<endl;
    return 0;
}
```

Дар ин ҷо массиви дученака муаррифӣ гардидааст, ки зернависи якуми он миқдори сатрҳо ва зернависи дуввуми он бошад, дарозии ҳар як сатрро нишон медиҳад, яъне дар мисоли мо дарозии ҳар як сатр то 15 рамз шуда метавонад. Натиҷаи барнома ба сурати зерин аст:

```
Toirov
Dodarov
Aliev
Saidov
Valiev
```

Равон кардани сатрҳо ба функсия ба сифати аргумент

Тавре ки медонем, сатрҳо дар забони C++ ҳамчун массивҳои рамзӣ фаҳмида мешаванд ва барои ба функсия ҳамчун аргумент равон кардани онҳо дар функсия ишорагаре ба номи сатри додашуда истифода бурда мешавад. Инчунин онҳо метавонанд монанди массивҳо ба функсия фиристода шаванд.

Мисол. Функсияе нависед, ки амали функсияи strlen-ро иҷро кунад.

```
#include <iostream>
#include <string>
using namespace std;
int Len(char *str){
    for (int s=0;*str;s++,str++)
        return s;
}
int main(){
    char str[]="This is string";
    cout<<"Darozii satr bo istifoda az tobei farii
Len="<<Len(str)<<endl;
    cout<<"Darozii satr bo istifoda az strlen="<<strlen(str)<<endl;
    return 0;
}
```

Дар ин ҷо функсияе ба номи Len муаррифӣ гардидааст, ки аргументи он як ишорагар бар сатре мебошад, сипас, дар дохили функсия тағйирёбандаи бутуни s муаррифӣ шудааст. Дар дохили ҳалқа то замони ба охири ҳалқа

расидан микдори рамзҳо ҳисоб карда шуда, бо ёрии оператори return баргардонида мешаванд. Дар функсияи аслии бошад, як маротиба функсияи фаръии навиштамон дар оператори хоричкунӣ даъват карда шудааст. Функсияи даъватшуда дарозии сатри додасударо ҳисоб намуда, қимати онро ба чоп медиҳад. Барои санҷиши дурусти функсия, як маротиба функсияи стандартии strlen низ истифода карда шудааст. Натиҷаи барнома ба сурати зерин аст:

```
Darozii satr bo istifoda az tobei farii Len=14
Darozii satr bo istifoda az tobei standardii strlen=14
```

Тавре ки аз натиҷаи барнома дида мешавад, қимати функсияи навиштамон ба қимати функсияи стандартӣ яхела аст.

Синфи нави кор бо сатрҳо дар забони C++

Дар фарзияҳои (версияҳои) нави забони C++ нави тағйирёбандаҳои сатрӣ, ки корбурдашон хеле сода аст, амал мекунад. Азбаски нави мазкур васеъшудаи навҳои пешина мебошад, бинобар ин, дар китобхонаҳои стандартӣ нигоҳ дошта мешавад. Агар ба забони техники гӯем, нави нав, ки string ном дорад, class мебошад. Class-ин нави додаҳо буда, дар барнома ё дар китобхонаи махсус нигоҳ дошта мешавад.

Нави char, ки қаблан истифода кардем, он чузъиёт ва методҳои классии string-ро надошта танҳо ишорагаре ба массивҳои нави рамзӣ буд. Сабаби камтар аз тағйирёбандаҳои классии string истифода кардан дар он аст, ки:

1. На ҳамаи компиляторҳои забони C++ классии string-ро пушти-бонӣ мекунад.

2. Барои ҳар як рамз тағйирёбандаҳои нави char ҳофиза ҷудо мекунад, вале тағйирёбандаҳои классии string ин хусусиятро надошта, ҳофизаи зиёдро банд мекунад.

Бар замми ин, набояд аз истифодабарии тағйирёбандаҳои нави string рӯй гардонид, зеро онҳо дорои методҳои зиёде ҳастанд, ки дар тағйирёбандаҳои нави char ин хусусият вучуд надошт.

Барои истифодабарии классии string бояд пеш аз ҳама, китобхонаи #include <string> насб карда шавад, ки классии string –ро пуштибонӣ мекунад. Китобхонаи мазкур аз китобхонае, ки функсияҳои нави char-ро пуштибонӣ мекард, фарқ мекунад. Истифодабаранда метавонад, дар як вақт ҳарду китобхонаро пайваст кунад, вале барои кор кардан ба додаҳои нави string пайваст намудани китобхонаи #include <string.h> лозим намебошад.

Сохтани сатр

Сатрҳо метавонанд бо якчанд услуб ё роҳ сохта шаванд. Масалан, ҳамчун сатри ҳолӣ, бо қимати аввала, бо дарозии муайян, иборат аз як қимати муайян, аз қимати сатри дигар ва ғайра. Мисол:

```
string sEmpty ; //сатри ҳолӣ
string str("In satr ast");
string s1="satri dodashuda";
string s2(str); // сохтани сатр дар асоси сатри дигар;
```

```
string sFill(10, 'A') ; // сатр иборат аз даҳто ҳарфи А;  
string str2(str,5, 4); // сатр иборат аз чорто рамзи сатри str, сар карда аз  
мавқеи панҷум.
```

Кор бо тағйирёбандаҳои навъи string

Класси string дорои як қатор имкониятҳо, методҳо ва хосиятҳои мебошад, ки онҳо дар тағйирёбандаҳои навъи char хос набуданд. Масалан, барои нусхабардорӣ намудани сатр, муқоисакунии ду сатр ва якҷоякунии ду сатр лозим нест, ки ягон функсияе даъват карда шавад. Масалан, бигзор шумо дорои ду сатр бошед.

```
string firstName (“Saidov”);  
string name (“Bobo”);
```

Ин тағйирёбандаҳо метавонанд ба ягон сатри дигар қимати худро бахшанд ва қимати он сатро дар худ гирд оваранд. Масалан, ба сатри firstName метавон ба сурати зерин қимати нав бахшид:

```
firstName=”Saidov”;
```

Барои муқоиса кардани сатрҳо метавон аз оператори муқоисакунии (==) истифода бурд. Ҳангоми муқоисакунии натиҷаи шарт ба қимати true баробар мешавад, агар сатрҳо якхела бошанд, дар ҳолати акс қимати шарт ба false баробар мегардад. (барои иҷрои ин амал рӯи тағйирёбандаҳои навъи char функсияи strcmp() даъват карда мешуд.)

```
if (name ==firstName)  
cout<<””name and firstName have the some name);
```

Барои нусхабардорӣ намудани як сатр ба сатри дигар аз оператори бахшиш ба сурати зерин истифода бурда мешавад.

```
string country=firstName;
```

Барои якҷоя кардани ду сатр низ аз оператори «+» ба сурати зерин истифода бурда мешавад.

```
string str= firstName+name;
```

Ин ифодаҳо метавонем низ ба сурати зерин навишт.

```
string str=firstName+” “ + name;
```

Дар ёд дошта бошед, ки истифодаи классии string табиати барномасозӣ ва кор кардан ба додаҳои сатриро тағйир надиҳад, танҳо имконияти истифодабарии додаҳои навъи сатриро васеътар месозад.

Доҳилкунӣ ва хориҷкунии додаҳои навъи string

Монанди дигар тағйирёбандаҳои навъҳои стандартии истифодабурдаамон, метавонем қимати тағйирёбандаҳои сатриро аз воридкунӣ ва ё натиҷаи тағйирёбандаҳои сатриро дар сафҳаи тасвир нависем. Объекти cin ва cout метавонанд барои доҳилкунӣ ва хориҷкунии додаҳои навъи string истифода бурда шаванд, вале ба сабаби он ки объекти cin то мавқеи фосилаи сатри додашударо хонда, қисми боқимондари сарфи назар мекунад, аз ин сабаб истифодаи он барои хондани сатрҳо маслиҳат дода намешавад. Барои бениҳад ворид намудани қимати тағйирёбандаҳои навъи мазкур аз функсияи getline() истифода мебаранд, ки нахви умумии он шакли getline(cin,str)-ро дорад:

Дар версияҳои пешина функсияи мазкур, ки барои тағйирёбандаҳои навъи char пешбинӣ шуда буд, қимати максималии миқдори сатрҳо низ ҳангоми ворид кардани сатр нишон дода мешуд, вале дар ин намуд бошад, метавон дилхоҳ сатрро ворид кард.

Мисол. Сохтан ва дохилкуниву хориҷкунии сатр бо истифода аз тағйирёбандаҳои навъи string.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string satr,name, firstName,Address,tel;
    cout<<"Enter you firstName and press Enter\n";
    getline(cin,firstName);
    cout<<"Enter you name and press Enter\n";
    getline(cin,name);
    cout<<" Enter you Address and press Enter \n";
    getline(cin,Address);
    cout<<"Enter you tel and press Enter\n";
    getline(cin,tel);
    satr="\nfirstName: "+firstName + "\nname: " +name+"\nAddress: "
+ Address + "\ntel: " + tel + "\n";
    cout<<satr;
    return 0;
}
```

Барои ҳалли мисоли мазкур аввал китобхонаи #include <string> ба барнома пайваст карда шудааст. Баъд аз ин, дар равиши барнома панҷ тағйирёбанда аз навъи string муаррифӣ шудаанд, ки мувофиқан чунин ном доранд:

```
string satr, name, firstName, Address, tel;
```

Баъд аз ин барои ворид кардани сатрҳои лозимӣ функсияи getline() ба сурати зерин даъват карда шудааст.

```
cout<<«Enter you firstName and press Enter\n»;
getline(cin,firstName);
cout<<«Enter you name and press Enter\n»;
getline(cin,name);
cout<<«Enter you Address and press Enter\n»;
getline(cin,Address);
cout<<«Enter you tel and press Enter\n»;
getline(cin,tel);
```

Тавре ки дар ин ҷо дида мешавад, дар функсияи getline миқдори максималии рамзҳо нишон дода нашудааст. Аз ин пас, сатри satr, ки аз қимати сатрҳои name, firstName ва Address,tel таркиб ёфтааст, дар охир бо ёрии объекти cout хориҷ карда шудааст. Натиҷаи барнома ба сурати зерин аст:

```
Enter you firstName and press Enter
Saidov
Enter you name and press Enter
Saidmurod
```

```
Enter you Address and press Enter
Dushanbe
Enter you tel and press Enter
93-454-44-44
firstName: Saidov
name: Saidmurod
Address: Dushanbe
tel: 93-454-44-44
```

Дастрасӣ ба символҳои тағйирёбандаҳои навъи string

Барои ба ҳар як рамзи сатр дастрас шудан, тағйирёбандаҳои сатриро ба намуди зернависдор навишта истифода мекунем. Ис-тифодаи нахви мазкур монанди тағйирёбандаҳои навъи char мебошад, яъне

Сатр[зернавис];

Масалан, барномаи дар поён овардашуда ҳар як рамзи сатри додасударо ҷудо карда, ҷоп мекунад.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string firstName="Saidov";
    for(int i=0;i<firstName.size() ;i++)
        cout<<firstName[i]<<endl;
    return 0;
}
```

Монанди тағйирёбандаҳои навъи char зернависи тағйирёбандаҳои навъи string низ аз нол оғоз мегардад. Шарти даврзании ҳалқа аз дарозии сатр вобастагӣ дорад. Барои дарозии тағйирёбандаҳои навъи char-ро дарёфт кардан аз функсияи length(), истифода бурда мешуд, барои дарозии тағйирёбандаҳои навъи string-ро дарёфт кардан аз методҳои классии string, ки нахви он сурати зеринро дорад,

сатр.size();

истифода бурда мешавад. Масалан, матни барномаи зерин ҳарфҳои сатри додасударо ба registerӣ болоӣ табдил медиҳад.

```
#include <iostream>
#include <string>
#include <ctype.h>
using namespace std;
int main(){
    string comp="Ilmhoi Computeri";
    for(int i=0;i<comp.size() ;i++)
        comp[i]=toupper(comp[i]);
    cout<<comp<<endl;
    return 0;
}
```

Амалиёт ба сатрҳо

Дар болои тағйирёбанда ё объектҳои навъи string операторҳои зерин метавонанд амал кунанд:

№	Оператор	Маъно ва мафҳум
1	=	Бахшиш
2	+	Якҷоякунӣ
3	==	Баробаркунӣ
4	+=	Якҷоякунӣ ва бахшидан
5	!=	Нобаробар
6	<	Хурд
7	<=	Хурд ва баробар
8	>	Калон
9	>=	Калон ва баробар
10	[]	Зернавис
11	>>	Дохилкунӣ
12	<<	Хориҷкунӣ

Мисол:

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str("In satr ast ");
    string s1="satri dodashuda";
    string s2=str+s1; // якҷоя намудани ду сатр
    cout<<s2<<endl;
    return 0;
}
```

Функсия-аъзоҳои кор бо сатрҳо

Монанди дигар контейнерҳо дар библиотекаи string барои коркарди сатрҳо функсияҳои зиёде мавҷуданд, ки муҳимтарини онҳо дар ҷадвали зерин гирд оварда шудаанд.

№	Номи функсия-аъзо	Маъно ва мафҳум
1	size()	Андозаи ҷорӣ сатрро муайян мекунад.
2	length()	Дарозии сатрро муайян мекунад
3	max_size()	Микдори максималии рамазҳои сатрро бармегардонад.
4	resize()	Андозаи сатрро тағйир медиҳад, яъне барои сатр андозаи нав барқарор мекунад.
5	reserve()	n-рамази сатрро бурида ме-гирад.
6	empty()	Қимати ҳақ бармегардонад, агар сатр холӣ бошад
7	erase()	nум рамази сатри додашударо сар карда аз мавқеи start нест мекунад.

8	append()	Як сатрро дар охири сатри дигар пайваст мекунад
9	find()	Мавқеи аввалини вукӯи зер-сатрро дар сатр муайян мекунад. (бармегардонад)
10	replace()	Барои рамзҳои сатри ҷориро бо рамзҳои зерсатр иваз намудани хизмат мекунад.
11	copy(char *s, unsigned int n, unsigned int m)	Аз мавқеи m-ум сар карда, n-рамзи сатри ҷориро ба масиви рамзи s нусхабардорӣ мекунад.
12	find_first_of()	Мавқеи аввалини рамз ё зерсатрро дар сатр ҷустуҷӯ мекунад.
13	find_first_not_of()	Мавқеи аввалини рамзи сатри ҷориро, ки ба рамзи зерсатр мувофиқат намекунад, муайян мекунад.
14	find_last_of()	Мавқеи охири рамз ё зерсатрро дар сатр ҷустуҷӯ мекунад.
15	find_last_not_of()	Мавқеи охири рамзи сатри ҷориро, ки ба рамзи зерсатр мувофиқат намекунад, муайян мекунад.
16	assign (const char *str, size_type num)	Барои як сатрро ба сатри дигар бахшидан истифода бурда мешавад.
17	at(unsigned int n)	Функсия аъзо буда, ба рамзи n-уми сатри ҷорӣ ишорагар бармегардонад.
18	substr(pos,n);	Аз мавқеи pos-ум сар карда n-рамзи сатри ҷориро бурида мегирад.
19	c_str	Ба масиви символие, ки ба симболи нол ба охир расида, қимати ҷорӣ объектҳои String мебошад ишорагар бармегар-донад.
20	swap	Қимати сатри ҷориро бо сатри додашуда иваз мекунад.
21	rbegin	Итератори баракс ба симболи охири сатр бармегардонад
22	push_back	Ба охири сатр симболи нав илова намуда, андозаи сатрро як воҳид зиёд мекунад.
23	pop_back	Симболи охири сатрро нест намуда, андозаи сатрро як воҳид кам мекунад.
24	clear	Маълумотҳои сатрро нест мекунад.
25	insert()	Пеш аз симболи нишондодашуда, символҳои нави лова мекунад.

Функсияи assign()

Функсияи мазкур барои як сатрро ба сатри дигар бахшидан, истифода бурда шуда, дорои ду қолаб мебошад. Дар поён нахвии умумии ин қолабҳо оварда шудаанд.

1. string &assign(const string &strob, size_type start, size_type num);
2. string &assign(const char *str, size_type num);

Қолаби якум num рамзи сатри strob –ро сар карда аз мавқеи start ба сатри даъватшаванда мебахшад. Ҳангоми истифодабарии қолаби дуввум, num рамзи сатри str-ро бо охиращасии сатр ба сатри даъватшаванда мебахшад. Дар ҳар ду қолаб ишорагар баргардонида мешавад. Барои як сатрро пурра ба сатри дигар бахшидан, аз оператори “=” истифода бурда мешавад. Функцияи assign() бошад, ҳамчун ҳолати хусусии оператори “=” истифода бурда мешавад, яъне он барои як қисмати як сатрро ба сатри дигар бахшидан хизмат мекунад. Мисол:

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string str("In satri kutoh ast.");
    string str2;
    str2.assign(str,2, 6);
    cout<<str2<<"\n";
    return 0;
}
```

Дар ин мисол str2 =satri мегардад.

Мисоли дигар. Ин мисол ҳар маротиба 5 рамзи зиёдтар рамзи сатри чориро ба сатри s нусхабардорӣ мекунад.

```
#include <iostream>
#include <string>
#include <ctype.h>
using namespace std;
int main()
{
    string s, str = "*****";
    int pos = str.length();
    while ( pos ){
        s.assign ( str.begin(),str.end() - pos+1);
        cout << s << endl;
        pos -= 5; }
    cout << endl;
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
*
*****
*****
*****
```

Функцияи append()

Функцияи мазкур барои як сатрро дар охири сатри дигар пайваст намудан, хизмат мекунад. Қолаби умумии он намуди зеринро дорад:

1. string &append(const string &strob, size_type start, size_type num);
2. string &append(const char *str, size_type num);

Қолаби якум num рамзи сатри strob –ро сар карда, аз мавқеи start ба қафои сатри даъватшаванда илова мекунад. Ҳангоми исти-фодабарии қолаби дуввум, num рамзи сатри str-ро бо охиррасии нол ба қафои сатри даъватшаванда илова мекунад. Дар ҳар ду қолаб ишорагар баргардонида мешавад.

Мисоли 1. Истифодаи функцияи append.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str("In satr ");
    string str2("kutoh ast.");
    str.append(str2,0, 10);
    cout<<str;
    return 0;
}
```

Мисоли 2. Барномаи зерин ҳар маротиба, як рамзи тағйирёбандаи рамзии ch-ро ба сатри str нусхабардорӣ мекунад.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str;
    for ( char ch = 'a'; ch <= 'z'; ch++ )
        str.append(1,ch);
    cout << "str is: " << str << endl;
    cout << endl;
    return 0;
}
```

Натиҷаи барнома чунин аст: str is: abcdefghijklmnopqrstuvwxyz

Функцияҳои insert(), erase() ва replace()

Функцияи erase() барои аз сатри додашуда, нест намудани num рамзро сар карда, аз мавқеи start истифода бурда мешавад. Қолаби умумии он намуди зеринро дорад:

```
string &erase(size_type start=0, size_type num=npos);
```

Функцияи insert() ва replace() барои иваз намудани рамзҳои сатрро ба рамзҳои дигар хизмат мекунанд. Қолаби умумии функцияҳои мазкур намуди зеринро дорад:

Функцияи insert:

1. string &insert (size_type start, const string &stob)
2. string &insert (size_type start, const string &stob, size_type insStart, size_type num);

Наҳви функцияи replace:

1. string &replace(size_type start, size_type num, const string &stob)
2. string &replace (size_type start, size_type orgNum, const string &stob, size_type replaceStart, size_typereplaceNum);

Қолаби якуми функцияи insert () num рамзи сатри strob-ро ба сатри даъватшаванда иваз мекунад. Қолаби дуввум бошад, num- рамзи сатри strob-ро аз мавқеи insStart ба мавқеи start-и сатри даъватшаванда мегузорад. Мисол:

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str = "C++ language";
    string s    = "is best ";
    int pos = 4;
    str.insert(pos,s);
    cout << "str is: " << str << endl;
    return 0;
}
```

Натиҷаи барнома чунин аст: C++ is best language

Қолаби якуми функцияи replace () барои num рамзи сатри strob-ро ба мавқеи start-и сатри даъватшаванда гузоштан истифода бурда мешавад. Қолаби дуввум бошад, orgNum рамзро дар сатри даъватшаванда аз мавқеи start иваз мекунад. **Мисол:**

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str1("In satr ");
    string str2("kutoh ast.");
    cout<<"str1="<<str1<<endl;
    cout<<"str2="<<str2<<endl;
    str1.insert(0,str2);
    cout<<"\nSatri str1 pas az guzoshtani symvol=\n"<<str1<<endl;
    str1.erase(0,str2.length());
    cout<<"\nSatri str1 pas az guzoshtani symvol=\n"<<str1<<endl;
    str1.replace(8,10,str2);
    cout<<"\nSatri str1 pas az guzoshtani symvol=\n"<<str1<<endl;
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
str1=In satr.
str2=kutoh ast.
satri str1 pas az guzoshtani symvol=
kutoh ast.In satr
satri str1 pas az guzoshtani symvol=
In satr
satri str1 pas az guzoshtani symvol=
In satr kutoh ast.
```

Мисоли дигар. Ин барнома аз мавқеи 5-ум сар карда, 15 рамзи сатри str-ро нест мекунад.

```
#include <iostream>
#include <string>
using namespace std;
```

```

int main()
{
    string str;
    for ( char ch = 'a'; ch <= 'z'; ch++ )
        str.append(1,ch);
    cout << "str is: " << str << endl;
    str.erase(4,15);
    cout << "Erased range from str : " << str << endl;
    cout << endl;
    return 0;
}

```

Натиҷаи барнома намуди зеринро дорад:

str is: abcdefghijklmnopqrstuvwxyz

Erased range from str : abcdtuvwxyz

Функсияи at()

Функсияи at дар мавқеи нишондодашуда, ишорагар ба символ бармегардонад. Наҳви умумии функсияи мазкур намуди зеринро дорад:

char& at (size_t pos);

const char& at (size_t pos) const;

Барномаи зерин, пас аз ҳар рамзи сатри додашуда, аломати хитоб «!» мегузорад.

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    string s = "Hello World";
    for ( int pos = 0; pos < s.length(); ++pos )
        cout << s.at(pos) << "!";
    cout << endl;
    return 0;
}

```

Натиҷаи барнома чунин аст: H!e!!!o! !W!o!r!!d!

Мисоли функсияи find_last_of()

Барои мавқеи охири рамз ё зерсатрро дар дохили сатр ҳус-туҷӯ кардан, истифода бурда мешавад. Қолаби умумии он шакли зеринро дорад:

```

size_t find_last_of (const string& str, size_t pos = npos) const;
size_t find_last_of (const char* s, size_t pos = npos) const;
size_t find_last_of (const char* s, size_t pos, size_t n) const;
size_t find_last_of (char c, size_t pos = npos) const;

```

дар ин ҷо:

str	сатри символҳо барои ҳустӯҷӯ;
pos	мавқеи охири символ дар сатр барои ҳустӯҷӯ;
s	ишорагари масиви символӣ;
n	миқдори қиматҳои символ барои ҳустӯҷӯ;
c	характери индивидуалӣ барои ҳустӯҷӯ;
size_t is	навъи бутуни беаломат.

Мисол:

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str("C++ is best language");
    string s = "g";
    cout << "str is:" << str << endl;
    cout << "s is: " << s << endl;
    int n = str.find_last_of(s);
    cout << "last_of '" << s << "' found" << " at position " << n+1 <<
endl;
    n = str.find_last_of(' ');
    cout << "last_of ' ' found" << " at position " << n+1 << endl;
    n = str.find_last_of(" la");
    cout << "last_of \" la\" found" << " at position " << n+1 << endl;
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

str is: C++ is best language

s is: g

last_of 'g' found at position 19

last_of ' ' found at position 12

last_of « la» found at position 18

Истифодаи функсияи find() ва copy()

Функсияи find() мавқеъ ё индекси аввалии вуқӯи зерсатрро дар дохили сатр муайян мекунад. Қолаби умумии он шакли зеринро дорад:

```
size_t find (const string& str, size_t pos = 0) const;
size_t find (const char* s, size_t pos = 0) const;
size_t find (const char* s, size_t pos, size_t n) const;
size_t find (char c, size_t pos = 0) const;
```

Функсияи copy() бошад, аз мавқеъи m-ум сар карда, n-рамзи сатри чориро ба масиви рамзии s нусхабардорӣ мекунад. Қолаби умумии функсияи мазкур намуди зеринро дорад:

```
size_t copy (char* s, size_t n, size_t pos = 0) const;
```

Мисол:

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str = "First Name: Ali";
    char fname[255];
    cout << "str is: " << str << endl;
    int n = str.find(':');
    str.copy(fname, n+1, 0);
    fname[n+1] = 0;
```

```

        cout << "fname is: " << fname << endl;

        return 0;
    }

```

Натиҷаи барнома чунин аст: fname is: First Name:

Функсияҳои empty() ва erase(char *s)

Функсияи erase(char *s) –барои аз сатри додашуда, рамзи дар параметри s-нишондодашуро нест кардан, истифода бурда мешавад. Қолаби умумии функсияи мазкур намуди зеринро дорад:

```

string& erase (size_t pos = 0, size_t len = npos);
iterator erase (iterator p);
iterator erase (iterator first, iterator last);

```

дар ин ҷо:

pos	мавқеи симболи аввала барои нест кардан;
len	миқдори символҳое, ки бояд нест карда шаванд;
p	итератор ба симbole, ки бояд нест карда шаванд;
first, last	итератор ба фосилаи [first,last) символҳое, ки бояд нест карда шаванд.
size_t	навъи бутуни беаломат.

Функсияи empty() бошад, функсияи мантиқӣ буда, хангоми расидан ба охири сатр кимати true бармегардонад. Қолаби умумии он намуди зеринро дорад: bool empty() const;

Мисол:

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    string str = "ABCDEFGH";
    while ( ! str.empty() ){
        cout << str << endl;
        str.erase(str.end()-1);
    }
    cout << endl;
    return 0;
}

```

Ин барнома ҳар маротиба аз тарафи рост як рамзи сатри str-ро то расидан ба охири сатр, нест мекунад. Натиҷаи барнома намуди зеринро дорад:

```

ABCDEFGH
ABCDEFG
ABCDEF
ABCDE
ABCD
ABC
AB
A

```

Функция `substr(pos,n)`

Барои дар сатри `chori`, сар карда, аз мавқеи `pos`-ум, чудо намуда гирифтани `n`-рамз, истифода бурда мешавад. Қолаби умумии функцияи мазкур намуди зеринро дорад:

```
string substr (size_t pos = 0, size_t len = npos) const;
```

Мисол:

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string str = "We go step by step to the target";
    cout << "str is: " << str << endl;
    int n = str.find("step");
    string s = str.substr(n);
    cout << "s is: " << s << endl;
    cout << endl;
    return 0;
}
```

Дар ин мисол, аввал тавассути функцияи `find()` мавқеи нахустини зерсатри `step` муайян карда мешавад. Баъд аз ин мавқеъ сар карда, то охири сатр рамзҳои сатри `chori` ба `S` нусхабардорӣ мекунад. Натиҷаи барнома чунин аст:
`s is: step by step to the target`

Агар функцияи мазкурро ба намуди `s = str.substr(6,12)`-даъват кунем, он гоҳ натиҷа чунин мешавад: `s is: step by step`

Функция `push_back`

Функцияи `push_back` барои дар охири сатр илова намудани символ истифода бурда мешавад. Пас аз илова намудани символ андозаи сатр як воҳид зиёд мешавад. Қолаби умумии функцияи мазкур намуди зеринро дорад:

```
void push_back (char c);
```

Барномаи зерин додаҳои як файли матнро хонда, дар сатри `S` захира мекунад.

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    string S;
    ifstream file ("test.txt",std::ios::in);
    if (file)
    {
        while (!file.eof()) S.push_back(file.get());
    }
    cout << S << '\n';
    return 0;
}
```

Функция swap

Функция swap барои сатри чориро бо сатри дар параметри str нишондодашуда иваз намудан хизмат мекунад. Қолаби умумии функцияи мазкур намуди зеринро дорад:

```
void swap (string& str)
```

Мисол. Иваз намудани мухтавои ду сатр.

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string s="first", s1="second";
    cout<<"s after swap="<<s<<endl;
    cout<<"s1 after swap="<<s1<<endl;
    s.swap(s1) ;
    cout<<"s befor swap="<<s<<endl;
    cout<<"s1 befor swap="<<s1<<endl;
    cin.get();
    return 0;
}
```

Натиҷаи барнома шакли зеринро дорад:

```
s after
swap==first
s1 after swap
=second
s befor
swap=second
s1 befor swap
=first
```

Функция c_str

Функция c_str ба массиви симболие, ки ба симболи нол ба охир расида, кимати ҷорӣи объектҳои String мебошад, ишорагар бармегар-донад. Қолаби умумии функцияи мазкур намуди зеринро дорад:

```
const char* c_str() const;
```

Мисол. Ба калимаҳо ҷудо намудани сатри додашуда.

```
#include <fstream>
#include <cstring>
#include <iostream>
#include <string>
using namespace std;
int main(){
    ifstream ifs;
    int ind, i, iter=0;
    string str;
    char s[5000];
    ifs.open("test.txt");
```

```

while (!ifs.eof() )
    ifs.getline(s, 5000);
str=s;
char * cstr = new char [str.length()+1];
strcpy (cstr, str.c_str());
char * p = strtok (cstr," ");
while (p!=0) {
    cout << p << '\n';
    p = strtok(NULL," ");
}
delete[] cstr;
return 0;
}

```

Дар ин ҷо аз унксияи strtok() истифода шудааст, ки он қисмати сатри додашударо то мавқеи нишондодашуда бармегардонад.

Функсияи rbegin

Функсияи rbegin() итератори баракс ба симболи охири сатр бармегардонад. Қолаби умумии функсияи мазкур намуди зеринро дорад:

```

reverse_iterator rbegin();
const_reverse_iterator rbegin() const;

```

Мисол. Баракс ҷоп намудани символҳои сатр.

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    string str ("Please help me");
    for (string::reverse_iterator rit=str.rbegin(); rit!=str.rend();
++rit)
        cout << *rit;
    return 0;
}

```

Натиҷаи барнома чунин аст: em pleh esaeIP

Функсияи clear

Функсияи clear() барои сатри додашударо аз маълумот холӣ намудан истифода бурда мешавад. Қолаби умумии ин функсия намуди зеринро дорад:

```

void clear()

```

Мисол.

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    char c;
    string str;
    cout << "Marhamt yak satr matn dokhil karda, baroi ba anjomrasi
nuqta(.)-ro zer kuned:\n";
    do {

```

```

        c = cin.get();
        str.push_back(c) ;
        if (c=='\n')
        {
            cout<<str;
            str.clear();
        }
    } while (c!='. ');
    return 0;
}

```

Функсияи pop_back

Функсияи pop_back симболи охирони сатрро нест намуда, андозаи сатрро як воҳид кам мекунад. Қолаби умумии ин функсия намуди зеринро дорад:

```
void pop_back()
```

Мисол.

```

#include <iostream>
#include <string>
using namespace std;
int main(){
    char c;
    string str ("Hello world!");
    str.pop_back();
    cout << str << '\n';

    return 0;
}

```

Саволҳо барои мустаҳкамкунӣ

1. Чӣ тавр тағйирёбандаҳои сатри муаррифӣ карда мешаванд?
2. Дар байни сатрҳо ва массивҳо чӣ тавр вобастагӣ вучуд дорад?
3. Сатрҳо бо ишорагарҳо чӣ тавр робита доранд?
4. Кадом функсияҳои стандартии кор бо сатрҳо мавҷуд аст?
5. Функсияи strcpy ва strcat чи вазифаро иҷро мекунанд?
6. Оё сатрҳо метавонанд ба сифати аргумент ба функсия раван карда шаванд?
7. Зернависи дуҷум дар массиви дученака чи амалро иҷро мекунад?
8. Тағйирёбандаҳои классии string чӣ гуна тағйирёбандаанд?
9. Бо кадом сабаб тағйирёбандаҳои классии string-ро камтар истифода мебаранд?
10. Оё тағйирёбандаҳои классии string имконияти аз тағйирёбандаҳои навъи char зиёдтарро доранд?
11. Чаро тағйирёбандаҳои навъи string-ро бо ёрии объекти cin ворид намекунанд?
12. Дарозии тағйирёбандаҳои классии string чӣ тавр муайян карда мешаванд?

13. Оё метавонем ҳар як рамзи тағйирёбандаҳои классии string –ро дар алоҳидаги барорем?

14. Тағйирёбандаҳои классии string чӣ тавр якҷоя, нусхабардорӣ ва муқоиса карда мешаванд?

15. Кадом методҳои классии string-ро медонед?

Намунаи ҳалли масъалаҳо

1. Сатр дода шудааст. Барномае нависед, ки миқдори ҳарфи «a»-и онро ҳисоб кунад.

```
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    int ln,m=0,i;
    char *b="adab oinaest bahri muaddab" ;
    ln=strlen(b);
    char c='a';
    for (i=0;i<ln;i++)
        if (b[i]==c)
            m++;
    cout<<"dar satri dodashuda "<<m<<" to harfi "<<c<<" mavjud
ast."<<endl;
    return 0;
}
```

2. Сатре дода шудааст. Барномае нависед, ки ҳарфи a-и онро бо ҳарфи «A» табдил диҳад.

```
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    int ln,m=0,i,k=0;
    char a[100],*b="adab oinaest bahri muaddab" ;
    ln=strlen(b);
    char c='a';
    for (i=0;i<ln;i++,k++)
        if (b[i]==c){
            m++;
            a[k]='A';}
        else {
            a[k]=b[i];
        }
    cout<<"satri asli="<<b<<endl;
    cout<<"satri tagiyrdodashuda="<<a<<endl;
    return 0;
}
```

3. Сатре, ки дар он адади бутун дар системаи ҳисоби дӯи дода шудааст. Барномае нависед, ки онро ба даҳӣ гардонад.

```
#include <iostream>
#include <string.h>
```

```

using namespace std;
int main(){
    char bin[16];
    long int dec=0;
    int i,v=1;
    cout<<"bin=";<<cin>>bin;
    for (i=strlen(bin)-1;i>=0;i--) {
        if (bin[i]=='1')
            dec+=v;
        v*=2;
    }
    cout<<"dec="<<dec;
    return 0;
}

```

4. Сатре дода шудааст, ки аз рақамҳои системаи шонздаҳӣ иборат мебошад. Барномае нависед, ки сатри додашударо ба системаи даҳӣ гардонад.

```

#include <iostream>
#include <string.h>
using namespace std;
int main(){
    char st[5];
    unsigned int dec=0;
    int i,v=1,er=0;
    cout<<"st=";<<cin>>st;
   strupr(st);
    for (i=strlen(st)-1;i>=0;i--){
        if (st[i]>='0' && st[i]<='9')
            dec+=v*(st[i]-48);
        else
            if (st[i]>='A' && st[i]<='F')
                dec+=v*(st[i]-55);
            else {
                er=1;
                break;
            }
        v*=16;
    }
    if(!er) {
        cout<<st<<endl;
        cout<<"dec="<<dec;
    }
    else {
        cout<<"st="<<endl;
        cout<<"error"<<endl;
        cout<<"dec="<<dec;
    }
    return 0;
}

```

5. Барномае нависед, ки адади бутуни додашударо аз системаи ҳисоби даҳӣ ба шонздаҳӣ табдил диҳад.

```
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    int dec,r,i=4;
    char st[5];
    cout<<"dec=";<<cin>>dec;
    st[5]='\0';
    do{
        r=dec % 16;
        dec=dec/16;
        if(r<10)
            st[i--]=r+48;
        else
            st[i--]=r+55;
    }
    while (dec>10);
    ++i;
    int j=0;
    while (st[i])
        st[j++]=st[i++];
    st[j]='\0';
    cout<<"st="<<st<<endl;
    return 0;
}
```

6. Номи n нафар донишҷӯ дода шудааст. Барномае нависед, ки бо истифода аз алгоритми мураттабсозии «хубобӣ» онҳоро аз рӯи алифбо мураттаб намуда, хорич кунад.

```
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    char *temp;
    int i,j;
    const int n=7;
    char *array[]=
{"Saidov","Mulloev","Saidov","Aliev","Valiev","Toirov","Abdulloev"};
    for (i=0;i<n;i++)
        for (j=0;j<n;j++)
            if (strcmp(array[i],array[j])<0){
                temp=array[i];
                array[i]=array[j];
                array[j]=temp;
            }
    for (i=0;i<n;i++)
        cout<<array[i]<<" \n ";
    return 0;}
}
```

7. Барномае нависед, ки натиҷаи имтиҳони донишҷӯёнро нишон ди-ҳад.

```
#include <iostream>
#include <string.h>
#include <stdio.h>
using namespace std;
int main(){
    int n[6],s=0;
    float p[6];
    char *mes[] ={"\0","\0","du\0","se\0","chor\0","panj\0"};
    int i;
    cout<<"korkardi natilai imtihon\n";
    cout<<"Qimathoro dohil kuned\n";
    for(i=5;i>=2;i--){
        printf("%s->",mes[i]);
        scanf("%i",&n[i]);
        s+=n[i];
    }
    for(i=2;i<6;i++)
        p[i]=(float)n[i]/s*100;
    printf("%8s %2s %5s%\n", "baho","miqdor","foiz");
    for(i=5;i>=2;i--
        printf("%8s %2i %6.0f%\n", mes[i],n[i],p[i]);
    return 0;
}
```

8. Сатре дода шудааст. Барномае нависед, ки бо дода шудани рақами тартибии элемент он элементро аз сатр хат занед.

```
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    int i,m;
    cout<<"m=";<<cin>>m;
    char *a="In matn ast.";
    for(i=m;i<strlen(a);a[i]=a[i+1],i++);
    cout<<a<<" "<<" ";
    return 0;
}
```

9. Функсияе нависед, ки рамзҳои сатри S1-ро аз n-умаш сар карда, ба миқдори m рамзро ба сатри S2 нусхабардорӣ кунад.

```
#include <iostream>
#include <string.h>
using namespace std;
void Str(char S1[],char S2[],int n, int m);
{
    int i,j;
    for (i=n,j=0;j<m;S2[j++]=S1[i++]);
    S2[j]='\0';
}
int main(){
```

```

char S1[25]="matematika",S2[25];
Str(S1,S2,4,8);
cout<<S2<<endl;
return 0;
}

```

Масъалаҳо барои кори мустақилона

1. Муайян кунед, ки кадом ҳарфи алифбо дар ҷумлаи додашуда истифода нашудааст.
2. Ҷумла дода шудааст, пас аз ҳар садонок аломати дефис «-» гузored, яъне ҷумларо ба ҳиччоҳо тақсим кунед.
3. Массиви $a[m]$ ва $b[n]$ -ро тавре тартиб диҳед, ки элементҳояшон мувофиқан садонокҳо ва ҳамсадоҳо бошанд.
4. Ифодае аз се калима иборат дода шудааст. Барномае нависед, ки ҷойи калимаи охири онро ва аввалии онро иваз кунад.
5. Дар ҷумлаи додашуда миқдори ҳарфи додашударо ёфта муайян кунед, ки чанд ҷисади ҳарфҳои боқимондаро ташкил медиҳанд.
6. Сатри матн дода шудааст. Барномае нависед, ки қимати миёнаи арифметикии ASCII-коди рамзҳои онро ҳисоб кунад.
7. Муайян кунед, ки сатри додашуда бо ҳарфи саршудааш ба итмом мерасад ё не.
8. Барномае нависед, ки ном ва рақами телефони 30 нафарро мувофиқан дар ду массив name ва tel хонда захира кунад. Сипас, бо ворид кардани номи ягон шахс рақами телефони вайро ёфта хориҷ кунад.
9. Дар ҷумлаи додашуда миқдори такроршавии ҳар калимаро муайян кунед.
10. Матн дода шудааст. Барномаи нависед, ки онро бо истифода аз усули кӯчиш рамзгузори кунад. Моҳияти ин усул чунин аст: Матн ва як адади бутуни фосилаи ($1 < n < 26$) дода мешавад. Ҳар як ҳарфи алифбо вобаста бо қимати n ба тарафи рост кӯчонида мешавад. Масалан, ҳангоми $n=3$ будан ҳарфи А ба D, ҳарфи D ба E, ҳарфи E ба F ва ҳоказо кӯчонида мешавад. Пас аз иҷрои ин амал, ба ҷои ҳарфи А ҳарфи D, ба ҷои D ҳарфи E, ба ҷои E ҳарфи F ва ғайра навишта мешавад.
11. Барномае нависед, ки матн бо усули кӯчиш рамзгузошташударо рамзкушоӣ кунад.
12. Сатр дода шудааст. Барномае нависед, ки миқдори ҳарфи «a»-и онро ҳисоб кунад.
13. Сатр дода шудааст. Барномае нависед, ки ҳарфи a-и онро бо ҳарфи «A» табдил диҳад.
14. Аз калимаи информатика ба воситаи амали «бурида гирифтани» ва «ҷаспондани» калимаи роман тартиб дода шавад.
15. Калимаи s дода шудааст. Калимаи нави $s1$, ки дар натиҷаи аз тарафи рост ба чап хондани калима s пайдо мешавад хориҷ карда шавад.
16. Сатре, ки дар он адади бутун дар системаи ҳисоби дӯи дода шудааст. Барномае нависед, ки онро ба системаи даҳӣ гардонад.

17. Сатр дода шудааст, ки аз рақамҳои системаи шонздаҳӣ иборат мебошад. Барномае нависед, ки сатри дода шударо ба системаи даҳӣ гардонад.
18. Барномае нависед, ки адади бутуни додашударо аз системаи ҳисоби даҳӣ ба шонздаҳӣ табдил диҳад.
19. Сатр дода шудааст. Барномае нависед, ки бо дода шудани рақами тартибии калима, калимаро аз сатр хат занед.
20. Зербарномае нависед, ки рамзҳои сатри S1-ро аз мавқеи n-умаш сар карда, ба миқдори m рамзро ба сатри S2 нусхабардорӣ кунад.
21. Ҷумла дода шудааст. Дар он ҳама рамзҳои дар мавқеи ҷойгиршавии сеюм, шашум, нӯҳум ва ғ. буда ба ҳарфи “a” иваз карда шаванд.
22. Калима дода шудааст. Агар миқдори ҳарфҳои калима чуфт бошад, ҳарфҳои дар мавқеи тоқ-ро ҷоп кунед. Ва агар тоқ бошад, ҳарфҳои дар мавқеи чуфт ҷоп карда шаванд.
23. Калимаи 12 - ҳарфа дода шудааст. Ҳарфҳои он чунин ҷобачогузорӣ карда шаванд: якум, дувоздахум, дуюм, ёздахум,..., панҷум, ҳаштум, шашум, ҳафтум.
24. Ду ҷумла дода шудааст. Муайян карда шавад, ки оё калимаҳое, ки ҷумлаи якумро ташкил додаанд дар ҷумлаи дуюм дида мешаванд.
25. Дар ҷумлаи додашуда калимаи якум бо калимаи охиринаи он иваз карда шавад.

Боби 9. Сохторҳо (structures)

Муқаддима

Дар боби массивҳо мо ба сохторҳои сарукур гирифтаем, ки онҳо аз маҷмӯи элементҳои якҷинса ва дорои сифати якхела таркиб ёфта буданд. Масалан, дастури `int a[25]` муаррифгари як маҷмӯи элементҳои бутун мебошад, ки дар он метавон 25 адади бутунро захира кард. Ба ҳамин монанд массивҳои навъҳои дигар муаррифӣ карда мешаванд. Акнун фарз мекунем, маҷмӯи додаҳои гуногуннавъе дода шуда бошанд. Барои ин навъ маҷмӯи додаҳо аз массивҳо истифода бурда намешавад, зеро ки массивҳо барои элементҳои аз ҷиҳати навъ якхела пешбинӣ шудаанд ва дар онҳо элементҳои якнавъ нигоҳ дошта мешаванд.

Масалан, мехоҳем иттилооти як нафар коргарро, ки иборат аз насаб, ном, синну сол ва маош мебошанд, дар зери як ном муаррифӣ кунем. Барои ин аз сохтмони додаҳои, ки сохтор ном доранд, истифода бурда мешавад. Сохторҳо маҷмӯи додаҳои мебошанд, ки ба онҳо дар зери як ном муроҷиат карда мешавад. Сохторҳо имконият медиҳанд, ки бо роҳи сода додаҳои гуногунавъ дар зери як ном гирд оварда шаванд. Ин навъ маҷмӯи додаҳо дар забонҳои Паскал, Cobol ва PL/I `record` номида мешаванд.

Ба ибораи дигар, сохторҳо навъи нави мебошанд, ки тавассути корбар муаррифӣ карда мешаванд. Додаҳои ин навъ як тағйирёбанда нубуда, балки метавонанд гурӯҳи тағйирёбандаҳо бошанд.

Пас аз пайдоиши БОН аз сохторҳо ҳарчанд, ки камтар истифода бурда мешавад, вале ба ин нигоҳ накарда, онҳо дар баъзе корбурдҳои истифодаи васеъ доранд. Омӯхтани сохторҳо заминаи омӯхтани БОН-ро фароҳам меорад.

Муаррификунии навъи сохтор

Барои истифода аз сохтмони додаҳои сохтор аввал навъи сохтор эҷод карда шуда, сипас тағйирёбандаҳои лозими аз ин навъ муаррифӣ карда мешаванд.

Наҳви умумии муаррификунии як сохтор шакли зеринро дорад:

```
struct struct_name
{
    Var1;
    Var2;
    ...
    Var n
};
```

Дар ин ҷо `struct`-калимаи калидӣ буда, `struct_name` номи сохтор, `var1`, `var2`, ..., `varn` аъзо ё тағйирёбандаҳои сохтор мебошанд, ки онҳоро майдон (`field`) низ меноманд. Дар дохили сохтор барои ҳар як майдон навъи он низ нишон дода мешавад. Барои номгузории сохторҳо, аз қонуни номгузории идентификаторҳо истифода бурда мешавад. Ҳамаи майдонҳои сохтор дар дохили қавси системавӣ навишта шуда, рамзи «;» интиҳо ё ба охираи сохторро муайян мекунанд.

Бояд қайд кард, ки сохтор пеш аз функсияи асосӣ муаррифӣ карда мешавад.

Пас аз он ки сохтор муаррифӣ шуд, тағйирёбандаҳои лозимиро метавон аз навъи он муаррифӣ кард.

Азoi сохтор метавонанд, тағйирёбандаҳои маъмулӣ, ишорагарҳо, массивҳо ва ҳатто сохторҳои дигар бошанд. Ба ғайр аз ин, номи аъзоҳои сохтор низ метавонанд, тағйирёбандаҳои бошанд, ки дар ҷойи дигар муаррифӣ ва истифода шудаанд, вале маслиҳат он аст, ки аз ин кор даст кашем. Сабабаш дар он аст, ки бо ин амал метавон гоҳе ба хатоги рубурӯ шуд.

Бигзор додаҳои як донишҷӯ ба сурати зерин дода шуда бошанд:

№	Насаб	Ном	Синну сол	Маош
1	Алиев	Вали	23	356,5

Аз ин ҷо сохтори донишҷӯ ба сурати зерин муаррифӣ карда мешавад:

```
struct Student {
    int no;
    string Fname;
    string Lname;
    int age;
    float balance;
};
```

Пас аз он ки сохтор бо номи Student муаррифӣ шуд, зарурати муаррифии тағйирёбандаҳои навъи сохтор пеш меояд, ки онҳо ба сурати зерин муаррифӣ карда мешаванд:

Student S1,S2,S3,...Sn;

Дар ин ҷо ба сифати навъ барои тағйирёбандаҳои S1, S2, S3,... Sn номи сохтор истифода шудааст. Тағйирёбандаҳои сохтор низ тавассути вергул аз ҳамдигар ҷудо карда мешаванд.

Илова бар ин, тағйирёбандаҳои навъи сохторро метавон ҳангоми муаррифкунии сохтор дарҳол муаррифӣ кард. Амали мазкур ба шакли зерин иҷро карда мешавад:

```
struct Student {
    int no;
    string Fname;
    string Lname;
    int age;
    float balance;
} S1,S2,S3,...Sn;
```

Ба ҳамин монанд, аз навъи сохтор метавон ба миқдори лозими тағйирёбанда муаррифӣ кард.

Илова бар ин, як сохтор метавонад дар дохили функцияи асосӣ муаррифӣ гардад. Ин навъ сохторҳо сохторҳои локалӣ меноманд.

Дастраси ба майдонҳои сохтор

Пас аз он, ки тағйирёбандаҳо аз навъи сохтор муаррифӣ шуданд, зарурати дастрасӣ ба онҳо пеш меояд. Ба монанди қимат бахшидан ба майдонҳо, дохил ва хориҷ намудани қимати майдонҳо ва ғайра. Тавассути тағйирёбандаи навъи сохтор ва оператори нуқта (dot) ба майдонҳои сохтор

дастрас шудан мумкин аст. Наҳви дастрасшави ба майдонҳои сохтор шакли зеринро дорад.

Тағйирёбандаи `_навъи_сохтор`, майдони `_сохтор`;

Ба ин тартиб барои дастрас шудан ба майдонҳои сохтори Student тавассути тағйирёбандаи S, ки қаблан муаррифӣ шуда буд, ба сурати зерин амал карда мешавад:

```
S.no;  
S.Fname;  
S.Lname;  
S.age;  
S.balance  
;
```

Ба ҳамин монанд низ метавон ба майдонҳои сохтори Student қимат бахшид:

```
S.no=1;  
S.Fname="Aliev";  
S.Lname="Vali";  
S.age=23;  
S.balance=123.54;
```

Ба ҳамдигар бахшидани сохторҳо

Монанди тағйирёбандаҳои муқаррарӣ як тағйирёбандаи сохторро метавон ба тағйирёбандаи дигаре аз ҳамин навъи сохтор бахшид. Масалан, фарз мекунем, ки тағйирёбандаҳои S ва S1 аз навъи сохтори Student мебошанд. Аз ин ҷо навишти дастурҳои зерин қобили қабул мебошад:

```
S.no=1;  
S.Fname="Aliev";  
S.Lname="Vali";  
S.age=23;  
S.balance=123.54;  
.....  
S1=S;
```

Пас аз ин майдонҳои тағйирёбандаи S1 низ дорои қимати ба майдонҳои S монанд мешавад.

Доҳил ва хорич намудани тағйирёбандаҳои сохтор

Пас аз муаррифӣкунии тағйирёбандаҳои навъи сохтор зарурати доҳил ва хорич намудани онҳо пеш меояд. Имконити яку якбора доҳил ва хорич намудани тағйирёбандаҳои сохтор вучуд надорад. Аз ин ҷо ҳар як майдони сохтор алоҳида-алоҳида доҳил ва хорич карда мешаванд.

Мисол. Барномае нависед, ки маълумоти n нафар донишҷӯро, ки дар сохтори Student муаррифӣ шудаанд, аз вуруди хонда, донишҷӯро, ки аз ҳама balance-и зиёдтар дорад, хорич кунад.

```

#include <iostream>
#include <string.h>
using namespace std;
struct Student {
    int no;
    char Fname;
    char Lname;
    int age;
    float balance;
};
int main(){
    Student S,max;
    int i, n;
    cin>>n;
    max.balance=-1;
    for (i=1; i<=n; i++)
    {
        S.no=i;
        cout<<"Enter Fname:"; cin>>S.Fname ;
        cout<<"Enter Lname:"; cin>>S.Lname ;
        cout<<"Enter age:"; cin>>S.age ;
        cout<<"Enter balance:"; cin>>S.balance ;
        if (S.balance>max.balance){
            max=S;
        }
    }
    cout<<max.no<<"\t"<<max.Fname<<"\t"<<max.Lname<<"\t"<<max.age<<"
\t "<<max.balance ;
    return 0;
}

```

Дар ин ҷо ду тағйирёбанда бо номи S ва max аз навъи Student муаррифӣ карда шудааст, сипас, ба майдони balance-и max қимати -1 бахшида шудааст. Пас аз ин, дар дохили ҳалқаи такрорӣ маълумоти n нафар донишҷӯ аз клавиатура дохил карда мешаванд. Пас аз ҳар дохилкунӣ қимати balance-и S ба қимати balance-и max муқоиса карда мешавад. Агар қимати balance-и S аз қимати balance-и max калон бошад, ҳамаи майдонҳои S ба max бахшида мешаванд. Пас аз охирафии давр ҳамаи маълумотҳои донишҷӯи мавриди назар хориҷ карда мешаванд.

Ба аъзоҳои сохтор бахшидани қимати аввала

Ҳангоми муаррифӣкунии сохтор якбора наметавон ба ягон аъзои он қимати аввала бахшид. Барои ба майдонҳои сохтор бахшидани қимати аввала тағйирёбандае аз навъи сохтор муаррифӣ карда мешавад ва сипас тавассути он ба тағйирёбандаҳои сохтор қимати аввала бахшида мешавад. Қимати ибтидоӣ бояд ба тартиби ҷойгиршавии аъзоҳои сохтор дар дохили қавси системавӣ навишта шавад. Ба барномаи зерин диққат диҳед:

```

#include <iostream>
#include <string.h>

```

```

using namespace std;
int main(){
    double x;
    struct rec {
        int a;
        float b;
        double c;
    };
    static struct rec d={25,0.5,3.1492};
    x=d.a+d.b+d.c;
    cout<<"x="<<x;
    return 0;
}

```

Дар ин ҷо `d` як тағйирёбандаи доимии навъи сохтор мебошад, ки бо ёрии он ба тағйирёбандаҳои сохтор қимати аввала бахшида мешавад, яъне ба аъзои якум 25 ба аъзои дуюм 0,5 ва ба аъзои охири 3,1492. Агар дар дохили функсияе, ки дар он сохтор муаррифӣ шудааст, ба аъзоҳои сохтор қимати аввала бахшем, навиштани калимаи калидии `struct` лозим намебошад. Аз ин ҷо метавонем бо таври зерин ба аъзоҳои сохтор қимати аввала бахшем:

```
static rec d={25,0.5,3.1492};
```

ва ҳатто метавонем калимаи `static`-ро низ истифода набарем, яъне `rec d={25,0.5,3.1492}` нависем.

Методҳо. Муаррифӣ намудани методҳо

Метод ин функсияе мебошад, ки дар дохили сохтор муаррифӣ гардида, метавонад амали муносибери анҷом диҳад. Тамоми сигнатурҳои метод айнан ба функсия монанд буда, фарқиаш дар он аст, ки метод тавассути тағйирёбандаҳои навъи сохтор даъват карда мешавад. Барои даъвати функсия бошад, амали алоҳидае вучуд надорад, яъне метод қисми сохтор буда, барои даъвати он бояд объект сохта шавад. Аз ин рӯ, методҳоро инчунин функсия - аъзо низ меноманд. Натиҷаи бадастомадаи метод аз намуди даъвати он вобастагӣ дорад. Наҳви умумии муаррифӣсозии метод шакли зеринро дорад:

```

Тип номи_метод(Рӯйхати аргументҳо )
{
    Тани метод
}

```

Барои пурратар шинос шудан ба ин матлаб мисолеро дида мебароем. Мисол. Сохторе дорои ду майдони ададӣ дода шудааст. Барномае нависед, ки сумаи майдонҳои онро ҳисоб кунад.

```

#include <iostream>
#include <string.h>
using namespace std;
struct Simple {
    float a;
    float b;
    float sum(){
        return a+b;
    }
}

```

```

    }
};
int main()
{
    Simple s;
    s.a=2.5;
    s.b=3.8;
    cout<<s.sum() ;
    return 0;
}

```

Дар ин ҷо бо номи Sum() методе муаррифӣ шудааст, ки суммаи майдонҳои сохторро бармегардонад. Дар функцияи асосӣ, дар дастгоҳи вурудӣ тавассути тағйирёбандаи s методи мазкур даъват карда шудааст.

Ба ҳамин монанд, метавон методҳои дигарро низ муаррифӣ кард.

Конструктор. Истифодаи конструктор

Конструктор монанди як функция буда, барои ба тағйирёбандаҳои сохтор бахшидани қимати аввала хизмат мекунад. Конструктор дорои хосиятҳои зерин мебошад:

1. Номи конструктор бояд ба номи сохтор мувофиқат кунад.

2. Конструктор қимат бармегардонад ва барои он тип муайян карда намешавад.

Конструктор метавонад дорои аргумент ва ё аргумент бошад.

Вазифаи асосии конструктор ҳангоми муаррифӣ намудани тағйирёбандаи навъи сохтор ба майдонҳои он қимати аввала мебахшад. Барои ба ин матлаб хубтар шинос шудан ба мисоли зерин диққат диҳед:

```

#include <iostream>
#include <string.h>
using namespace std;
struct Student {
    int no;
    string Fname;
    string Lname;
    int age;
    float balance;
    //Конструктор
    Student(){
        no=1;
        Fname="Aliev";
        Lname="Vali";
        age=18;
        balance=1234.5;
    }
};
int main()
{
    Student S;
    cout<<S.no<<"\t"<<S.Fname<<"\t"<<S.Lname<<"\t"<<S.age<<" \t

```

```

"<<S.balance ;
    return 0;
}

```

Тавре айён аст, дар ин мисол конструкторе, ки номи он бо сохтор якхела аст, муаррифӣ гардидааст. Ин конструктор ҳамчун қимати ибтидои ба майдонҳои класс қимати ибтидои мебахшад. Сипас, дар методи асосӣ тағйирёбандаи S аз навъи Student муаррифӣ гардида, тавассути он майдонҳои сохтор хориҷ уарда шудаанд.

Илова бар ин, конструктор метавонад дорои аргумент бошад. Дар ин сурат, пас аз тағйирёбандаҳои навъи сохтор дар дохили қавси оддӣқимати майдонҳо навишта мешаванд. Ба мисоли зерин диққат диҳед.

```

#include <iostream>
using namespace std;
struct Student {
    int no;
    string Fname;
    string Lname;
    int age;
    float balance;
    Student(int n,string s, string s1, int a, float b)
    {
        no=n;
        Fname=s;
        Lname=s1;
        age=a;
        balance=b;
    }
};
int main(){
    Student S(1, "Aliev", "Vali", 23, 234.5);
    cout<<S.no<<"\t"<<S.Fname<<"\t"<<S.Lname<<"\t"<<S.age<<"\t "
<<S.balance ;
    return 0;
}

```

Массивҳо аз навъи сохтор

Массивҳоро метавон аз навъи низ сохтор муаррифӣ кард. Барои ин аввал сохтор ва баъд аз навъи он массив муаррифӣ карда мешавад. Элементҳои чунин массив ҳамон аъзоҳои сохтор мебошанд.

Мисоли 3. Барномае нависед, ки рақами тартибӣ, ному насаб, синну сол ва маоши 50 нафар коргарро барорад. Барои ин аз сохторҳо истифода баред, яъне рақами тартибӣ, фамилия, ном, синну сол ва маъош аъзоҳои сохтор бошанд.

```

#include <iostream>
using namespace std;
int main() {
    struct rec
    {

```

```

        char firstName[20];
        char name[20];
        int Ss;
        float maosh;
    } ;
    static struct rec d[]={
        {"Saidov","Said",22,650.0},
        {"Dodarov","Habib",19,649.9} ,
        {"Saidov","Salim",29,254.8} ,
        {"Younusov","Yousuf",23,120.5} ,
        {"Aminov","Amina",53,454.0} ,
    };
    cout<<"Rt  "<<"firstName\t"<<"\tname\t"<<"Age\t"<<"maosh\n";
    for (int i=0;i<4;i++)
        cout<<i+1<<"  "<<d[i].firstName<<"\t"<<d[i].name<<"\t"<<
d[i].Ss<<"\t"<<d[i].maosh<<endl;
    return 0;
}

```

Дар ин ҷо сохторе бо номи rec муаррифӣ шудааст, ки аъзоҳои он тағйирёбандаҳои Rt, firstName, name, Ss ва maosh мебошанд, яъне

```

struct rec{
    char firstName[20];
    char name[20];
    int Ss;
    float maosh;
} ;

```

Баъд аз ин, масиве бо номи d, ки андозаи он пешакӣ муайян нест, аз навъи сохтор муаррифӣ шуда, ба сифати қимати аввала ба элементҳои он қимати аъзоҳои сохтор бахшида шудаанд, яъне

```

static struct rec d[]={
    {"Saidov","Said",22,650.0},
    {"Dodarov","Habib",19,649.9} ,
    {"Saidov","Salim",29,254.8} ,
    {"Younusov","Yousuf",23,120.5} ,
    {"Aminova","Amina",53,454.0} ,
};

```

Дар ин мисол индекси калонтари массив баробари 3 мебошад, чунки массив дорои чор сатр аст ва аз 0 то 3 рақамгузори карда мешавад. Агар пешакӣ қимати сатрҳо муайян бошанд, метавонем дар вақти муаррифӣкунӣ андозаи массивро нависем, яъне d[3]. Дар қисмати баъдӣ бо ёрии оператори хоричкунӣ элементҳои массив, ки аз навъи сохтор буданд, хорич дода шудаанд.

```

for (int i=0;i<4;i++)
    cout<<i+1<<"\t"<<d[i].firstName<<"\t"<<d[i].name<<"\t"<<
d[i].Ss<<"\t"<<d[i].maosh<<endl;

```

Натиҷаи барнома шакли зайлро соҳиб мешавад.

Rt	firstName	name	Age	maosh
1	Saidov	Said	22	650
2	Dodarov	Habib	19	649.9

3	Saidov	Salim	29	258.9
4	Younusov	Yousuf	23	120.8

Таҳлили як сохтор

Ҳар як аъзои сохтор додаест мустақил ва метавон бо ҳар як аъзо амали алоҳидаеро анҷом дод. Барои дастрасшавӣ ба аъзоҳои сохтор аз оператори (dot) истифода бурда мешавад. Масалан, бигзор сохтори зерин муаррифӣ шуда бошад:

```
struct Time {
    int soat;
    int minute;
    int second;
};
```

Пас аз ин, бигзор тағйирёбандаи t-ро аз навъи сохтор муаррифӣ кунем. он гоҳ аз оператори зерин истифода мебарем.

Time t;

Барои қимате аъзоҳои сохторро аз клавиатура ворид кардан аз оператори dot(.) ба сурати зерин истифода бурда мешавад.

```
cin>>t.soat;
cin>>t.minute;
cin>>t.secont
```

Тавре ки қайд кардем, ҳар як аъзои сохтор мустақил буда, аз дигар аъзоҳо вобастагӣ надорад ва мо метавонем, болои ҳар кадом аъзо амали ҷудоғонаеро анҷом диҳем. **Мисол.** Сохтор дода шудааст, барои ҳар кадом аъзои он амали муайянеро анҷом диҳед.

```
#include <iostream>
using namespace std;
int main(){
    struct pard {
        int a;
        float f;
        double d;
    };
    pard P;
    cout<<"a=";cin>>P.a ;
    cout<<"f=";cin>>P.f ;
    cout<<"d=";cin>>P.d;
    P.a++;
    P.f+=22;
    P.d--;
    cout<<"a="<<P.a<<"\tf= " <<P.f <<"\t d= " <<P.d;
    return 0;
}
```

Тавре ки дида мешавад, дар ин мисол барои ҳар як аъзои сохтор амали муайяне иҷро шудааст, яъне барои аъзои бутуни сохтор амали инкремент, барои аъзои ҳақиқии сохтор амали ҷамъи унарӣ ва барои аъзои ҳақиқии калони

сохтор бошад, амали декримент ичро карда шуда, сипас бо ёрии оператори хоричкунӣ қиматҳои аъзоҳо хорич карда шудаанд.

Сохторҳои дохилиҳам

Ба ғайр аз ин, як майдони сохтор метавонад, ки худ сохтор бошад. Ин навъ сохторҳо сохторҳои дохилиҳам мегӯянд. Дар ин ҳолат бояд аввал сохтори беруна ва баъд сохтори даруна муаррифӣ гардад.

Мисол. Барномае нависед, ки ному насаб, рақами тартибӣ, рақами телефон ва суроғи донишҷӯро аз рӯи алифбо мурағаб намуда, ба ҳамчун ҳисобот хорич кунад. Ба ғайр аз ин, дар барнома имконияти ҷустуҷӯи маълумот дар бораи як донишҷӯ аз рӯи рақами тартибии он мавҷуд бошад, яъне рақами тартибии як донишҷӯ аз оператори хоричкунӣ дохил карда мешавад, агар донишҷӯ бо ин рақам мавҷуд бошад, тамоми маълумотҳои он хорич карда шаванд ва дар сурати акс пайғоми муносибе хорич гардад. Маълумоти як донишҷӯ ба сурати зерин дода мешаванд:

Rt	Ном (name)			Телефон (tel)	Суроға (address)
	Фамилия (lastname)	Ном (firstname)	Насаб (middle name)		

Тавре ки аз ин чадвал дида мешавад, додаҳои як донишҷӯ сохторест, ки дорои чор майдон `rt`, `name`, `tel` ва `address`. Илова бар ин ҷо боз майдони `name` сохторест, ки дорои ду узв. Барои ҳалли ин масъала аз сохторҳои дохилиҳам истифода мебарем. Агар номи сохтори аслиро `donishJu` гузорем, пас метавонем, ба ҳар як аъзои он ба сурати: `donishJu.rt`, `donishJu.name`, `donishJu.tel` ва `donishJu.address` муроҷиат кунем. Ба ҳамин монанд барои ба аъзоҳои сохтори `name` дастрас шудан аз оператори дот ду маротиба истифода карда мешавад. Барои сохтори `name` номи ихтисос медиҳем. Масалан, `name1` ва сипас бо аъзоҳои он ба шакли зерин дастрас шудан мумкин аст.

```
donishJu.name1.lastname,
donishJu.name1.firstname
donishJu.name1.middle.name
```

Ҳал. Коди барномаи мавриди назар ба сурати зерин аст.

```
#include <iostream>
using namespace std;
int main() {
    struct rec {
        int st_no;
        struct name {
            char l_name[25];
            char f_name[25];
            char m_name[25] ;
        }name1;
        char tel[15];
        char addr[30];
    };
    int i,j,n,s_no;
```



```

struct rec donishJu[70],temp;
cout<<"n=";cin>>n;
for(i=0;i<n;i++) {
    cout<<"st_no=" ;    cin>>donishJu[i].st_no;
    cout<<"l_name=" ;    cin>>donishJu[i].name1.l_name ;
    cout<<"f_name=" ; cin>>donishJu[i].name1.f_name ;
    cout<<"m_name=" ;    cin>>donishJu[i].name1.m_name ;
    cout<<"tel=" ;    cin>>donishJu[i].tel ;
    cout<<"addr=" ;    cin>>donishJu[i].addr ;
}
cout<<"Murattab nishon dodani donishjuyon az rui raqam\n";
for (i=0;i<n;i++)
    for (j=i+1;j<n;j++)
        if(donishJu[i].st_no>donishJu[j].st_no) {
            temp=donishJu[i];
            donishJu[i]=donishJu[j];
            donishJu[j]=temp;
        }
    cout<<"st_no "<<" firstNameilya\t"<<" name\t" <<"
LastName "<<" telphon\t" <<" Address\n" ;
    for (i=0;i<n;i++) {
        cout<<donishJu[i].st_no<<"\t";
        cout<<donishJu[i].name1.l_name<<" ";
        cout<<donishJu[i].name1.f_name<<" " ;
        cout<<donishJu[i].name1.m_name<<" " ;
        cout<<donishJu[i].tel<<" " ;
        cout<<donishJu[i].addr <<" " ;
        cout<<endl;
    }
    cout<<"\n raqami tartibii donishjuro baroi justuju
vorid kuned"<<endl;
    cout<<"st_no="; cin>>s_no;
    for (i=0;i<n;++i)
        if(s_no== donishJu[i].st_no)
            break;
    if (i<n) {
        cout<<"firstName="<<donishJu[i].name1.l_name<<endl;
        cout<<"name="<<donishJu[i].name1.f_name<<endl;
        cout<<"LastName="<<donishJu[i].name1.m_name<<endl;
        cout<<"tel="<<donishJu[i].tel<<endl;
        cout<<"Address="<<donishJu[i].addr <<endl;
    }
    else
        cout<<"Bo in raqam donishju vujud nadorad!!"<<endl;
    return 0;
}

```

Ишорагар ба сохтор (pointer to struct)

Дар забони C++ ишорагар ба сохтор монанди ишорагар ба дигар навъҳои стандартӣ мебошад.

Монанди ишорагарҳои дигар барои муаррифи намудани ишорагар ба сохтор аз оператори ситорача (*) истифода бурда мешавад. Оператори мазкур дар вақти муаррификунӣ пеш аз номи ишорагар навишта мешавад. Масалан, барои муаррифи намудани ишорагар барои класи Student, ки пештар муаррифи ва истифода шуда буд, дастури зерин кифоягӣ мекунад.

```
struct Student *addr_pointer;
```

Ишорагар ба сохторҳо дар ду ҳолат истифода бурда мешавад:

- 1) Барои ба сифати аргументи функция равон кардани суроғи сохтор.
- 2) Кор бо додаҳои динамикӣ.

Барои дастрас намудани суроғи тағйирёбандаи сохтор аз оператори & истифода бурда мешавад, ба китъаи барномаи зерин диққат диҳед.

```
char firstName[20];  
char name[20];  
int Ss;  
float maosh;  
struct Student *p; /* муаррифи ишорагар ба сохтор*/
```

Барои суроғи сохтори Student -ро ба тағйирёбандаи ишорагари p бахшидан аз дастури зерин истифода бурда мешавад:

```
p = &firstName;
```

Барои тавассути ишорагар дастрас шудан ба ягон аъзои сохтор бошад, аз оператори “->” истифода бурда мешавад, ки нахви он шакли зеринро дорад:

ишорагар->майдон.

Барои пурратар ба ин матлаб шинос шудан барномаи зеринро дида мебароем.

```
#include <iostream>  
using namespace std;  
struct adad{  
    int I;  
    float f;  
    char chr;  
};  
int main() {  
    adad a,b;  
    adad *x,*y;  
    x=&a; y=&b;  
    x->I=25;  
    x->f =2.5;  
    x->chr ='a';  
    (*y).I =12;  
    (*y).f =3.14;  
    (*y).chr='b';  
    cout<<"I="<<x->I <<endl;  
    cout<<"f="<<x->f<<endl;  
    cout<<"chr="<<x->chr<<endl<<endl;
```

```

cout<<"I="<<(*y).I <<endl;
cout<<"f="<<(*y).f <<endl;
cout<<"chr="<<(*y).chr<<endl;
return 0;
}

```

Тавре ки дида мешавад, дар ин ҷо сохторе муаррифӣ гардидааст, ки дорои се аъзо мебошад, яъне

```

struct adad{
    int I;
    float f;
    char chr;
};

```

Баъд аз ин, дар функсияи аслии ду тағйирёбандаи оддӣ ва ду тағйирёбандаи ишорагар аз навъи сохтор муаррифӣ гардидаанд. Дар тағйирёбандаҳои ишорагар суроғаи он ду тағйирёбандаи оддӣ ҷой дода шуда, сипас бо ду тарз ба аъзоҳои сохтор қимате бахшида шудааст.

```

x->I=25;
x->f =2.5;
x->chr ='a';

```

Дар ин тарз оператори “->” истифода бурда шудааст. Баъд аз ин, бо ёрии оператори ишорагар ва оператори dot бори дигар ба аъзоҳои сохтор қимат бахшида шудааст.

```

(*y).I =12;
(*y).f =3.14;
(*y).chr='b';

```

Баъд аз ин қиматҳои додашуда хориҷ карда шудаанд.

```

cout<<"I="<<x->I <<endl;
cout<<"f="<<x->f<<endl;
cout<<"chr="<<x->chr<<endl<<endl;
cout<<"I="<<(*y).I <<endl;
cout<<"f="<<(*y).f <<endl;
cout<<"chr="<<(*y).chr<<endl;

```

Натиҷаи барнома шакли зеринро дорад:

```

I=25
f=2.5
chr=a
I=12
f=3.14
chr=b

```

Равон кардани аъзоҳои сохтор ба функсия

Ҳар як майдони сохтор ё як сохторро ба таври комил метавон ба сифати параметр ба зербарномаҳо равон кард ва ё ҳамчун қимати интиҳои аз зербарнома ба функсияи асосӣ баргардонд. Равон кардани майдонҳои сохтор ба зербарнома ва баракс монанди тағйирёбандаҳои маъмулӣ иҷро карда мешавад.

Барои ин матлабро беҳтар фаҳмидан ва ба моҳияти он сарфаҳи рафтан мисолеро дида мебароем.

```
struct SimpleStruct{
    char x;
    int y;
    float z;
    char s[10];
} mike;
```

Пас аз муаррифӣ гардидани сохтор барои ба сифати аргументи зербарнома равон карлани аъзоҳо яке аз дастурҳои зерин истифода бурда мешавад:

```
func(mike.x); /* Қимати симболии x равон карда мешавад */
func2(mike.y); /* Қимати аъзои бутуни y равон карда мешавад */
func3(mike.z); /* Қимати аъзои ҳақиқии z равон карда мешавад */
func4(mike.s); /* Суроғаи аъзои сатрии s равон карда мешавад. */
func(mike.s[2]); /* Қимати симболи s[2] равон карда мешавад. */
```

Дар ҳар як аз ин ҳолатҳо ба функсия қимати элементи муайян равон карда мешавад.

Баъзан лозим мегардад, ки суроғаи элементи муайяне равон карда шавад, барои иҷрои ин амал пеш аз номи аъзои сохтор оператори & навишта мешавад. Ба сифати мисол ба дастурҳои зерин диққат диҳед:

```
func(&mike.x); /* Суроғаи симболии x равон карда мешавад */
func2(&mike.y); /* Суроғаи аъзои бутуни y равон карда мешавад */
func3(&mike.z); /* Суроғаи аъзои ҳақиқии z равон карда мешавад */
func4(mike.s); /* Суроғаи аъзои сатрии s равон карда мешавад. */
func(&mike.s[2]); /* суроғаи симболи s[2] равон карда мешавад. */
```

Тавре ки дида мешавад, симболи & пеш аз номи сохтор навишта мешавад, на пеш аз аъзоҳо. Ба ғайр аз ин, s худаш суроғаи сатрро муайян мекунад, аз ин ҷо пеш аз ин аъзо оператори & навишта намешавад.

Мисоли дигар. Барномае нависед, бо истифода аз зербарнома суммаи ду адади комплексӣ ҳисоб карда шаванд.

Қайд. *Аз нуқтаи назари риёзӣ медонем, ки адади комплекси намуди зеринро дорад:*

$$z = x + iy$$

дар ин ҷо $x=Re(z)$ – қисми ҳақиқии адади комплексӣ буда, $y=Im(z)$ – қисми мавҳум мебошад. Барои ҷамъ намудани ду адади комплексӣ қисмҳои ҳақиқӣ бо ҳақиқӣ ва қисмҳои мавҳум бо мавҳум ҷамъ карда мешаванд.

```
#include <iostream>
using namespace std;
struct Complex {
    float Re;
    float Im;
};
void InputComplex(Complex &X){
    cout<<"Enter Real part, unReal part \n";
    cin>>X.Re>>X.Im ;
```

```

}
int main()
{
    Complex x,y,z;
    InputComolex(x);
    InputComolex(y);
    freopen("output.txt","w",stdout);
    z.Re =x.Re+y.Re ;
    z.Im =x.Im +y.Im ;
    cout<<"Summa Real part="<<z.Re <<" Summ unreal part="<<z.Im ;

    return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

Enter Real part, unReal part
1.5  3
-2.8 1.3
Summa Real part=-1.3 Summ unreal part=4.3

```

Ишорагари сохторро баргардонидани функсия

Як функсия метавонад, ишорагареро ба як сохтор баргардонад. Мисоли зерин ин амалро нишон медиҳад.

Мисол. Барномае нависед, ки бо даъват кардани функсияи бо номи search массиве аз сохторро, ки аъзоҳои он маълумотҳои донишҷӯён мебошанд, баргардонад. Ин барнома бояд дорои чунин имкониятҳо бошад: бо дохил намудани рақами тартибии як нафар донишҷӯй маълумотҳои боқимондаи донишҷӯйро барорад. Агар рақами воридшуда ба рақами тартибии ягон донишҷӯ мувофиқат накунад, пайгоми муносибе хориҷ кунад. Амали дохил намудани қимат барои ҷустуҷӯ то ҳангоме, ки адад ғайринулӣ аст, идома ёбад ва ҳангоми ворид намудан сифр барнома кори худро қатъ намояд.

```

#include <iostream>
#include <string>
using namespace std;
#define Null 0
const int n=3;
struct student {
    int st_no;
    char *l_name;
    char *f_name;
    char *m_name;
    long recordbook;
    char *tel;
};
student *search(student table[],int s_no){
    int k;
    for(k=0; k<n;k++)
        if(table[k].st_no ==s_no)
            return (&table[k]);
}

```

```

        return Null;
    }
int main() {
    static student customer[]={
        {1,"Saidov","Umar","Kurbonovich",254,"918-88-88-88"},
        {2,"Dodarov","Habib","Kurbonovich",124,"93-333-33-33"},
        {3,"Aliev","Vali","Saidovich",1201,"918-12-34-56"},
    };
    int s_no;
    student*pt;
    cout<<"s_no=";<<cin>>s_no;
    while( s_no!=0){
        pt=search(customer, s_no);
        if(pt!=Null) {
            cout<<"l_name="<<pt->l_name <<endl;
            cout<<"f_name="<<pt->f_name <<endl;
            cout<<"m_name="<<pt->m_name <<endl;
            cout<<"recordbook="<<pt->recordbook<<endl;
            cout<<"tel="<<pt->tel<<endl;
        }
        else
            cout<<"\nerror-pleas try again!!\n";
        cout<<"\ns_no=";<<cin>>s_no;
    }
    return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

s_no =1
l_name=Saidov
f_name=Umar
m_name= Kurbonovich
recordbook=254
tel=918-88-88-88

s_no=5
error-pleas try again!!
s_no=0

```

Ба сифати аргументи функция равон кардани сохтор

Метавон тамоми як сохторро ба функция ба сифати аргумент низ фиристонд ва ё як сохторро аз функция бо ёрии оператори return баргардонид. Чунин навъ равон кардани сохтор аз рӯи суроға набуда, аз рӯи миқдор мебошад, яъне як сохтор бо тамоми нусхабардорӣ шуда, ба функция равона карда мешавад. Ҳар амале, ки дар функция болои ин сохтор иҷро мекунад, дар функцияҳои дигар ҳатто функцияи аслии низ шинохта намешавад. Танҳо

тағйирот дар нуктаи фурӯхонанда фаҳмида мешаваду халос. Дар мисоли поён истифодаи ин амал нишон дода шудааст. **Мисол.**

```
#include <iostream>
#include <string>
using namespace std;
struct student {
    int st_no;
    char *l_name;
    char *f_name;
    char *m_name;
    long recordbook;
    char *tel;
};
void func(student inf)
{
    inf.st_no =1;
    inf.l_name ="Saidov";
    inf.f_name ="Umar";
    inf.m_name ="Kurbonovich";
    inf.recordbook =010201;
    inf.tel ="918-88-88-88";
}

int main(){
    static student inform={2,"Dodarov","Habib","Kurbonovich",124,"93-333-33-33"};
    cout<<"st_no="<<inform.st_no <<endl;
    cout<<"l_name="<<inform.l_name <<endl;
    cout<<"f_name="<<inform.f_name <<endl;
    cout<<"m_name="<<inform.m_name<<endl;
    cout<<"recordbook="<<inform.recordbook<<endl;
    cout<<"tel="<<inform.tel <<endl<<endl;
    func(inform);
    cout<<"st_no="<<inform.st_no <<endl;
    cout<<"l_name="<<inform.l_name <<endl;
    cout<<"f_name="<<inform.f_name <<endl;
    cout<<"m_name="<<inform.m_name<<endl;
    cout<<"recordbook="<<inform.recordbook<<endl;
    cout<<"tel="<<inform.tel <<endl<<endl;
    return 0;
}
```

Барномаи мазкур як сохторро нусхабардорӣ намуда, ба функсия равон мекунад. Функсия бошад, сохтори фиристодашударо дарёфт карда, ба он қимат мебахшад. Аз сабаби онки дар ин ҷо ба функсияи асли чизе баргардонида нашудааст, натиҷаи барнома фақат додаҳои дохили худӣ сохтор мебошанд. Қимате, ки дар функсия ба аъзоҳои сохтор бахшида шуда буд, дар функсияи асли шинохта намешавад. Натиҷаи барнома шакли зеринро дорад:

```

s_no =1
l_name=Dodarov
f_name=Habib
m_name= Kurbonovich
recordbook=124
tel=93-333-33-33

s_no=5
l_name=Dodarov
f_name=Habib
m_name= Kurbonovich
recordbook=124
tel=93-333-33-33

```

Илова бар ин, як функсия метавонад, ба сифати натиҷа як сохторро баргардонад. Барои намуна барномаи қаблро камтар тағйир медиҳем.

```

#include <iostream>
#include <string>
using namespace std;
struct student {
    int st_no;
    char *l_name;
    char *f_name;
    char *m_name;
    long recordbook;
    char *tel;
} ;
student func(student inf){
    inf.st_no =1;
    inf.l_name = "Saidov";
    inf.f_name = "Umar";
    inf.m_name = "Kurbanovich";
    inf.recordbook =010201;
    inf.tel = "918-35-36-61";
    return (inf);
}
int main(){
    static student inform={
        2, "Dodarov", "Habib", "Kurbanovich", 124, "93-333-33-33"};
    cout<<"st_no="<<inform.st_no <<endl;
    cout<<"l_name="<<inform.l_name <<endl;
    cout<<"f_name="<<inform.f_name <<endl;
    cout<<"m_name="<<inform.m_name<<endl;
    cout<<"recordbook="<<inform.recordbook<<endl;
    cout<<"tel="<<inform.tel <<endl<<endl;
    inform= func(inform);
    cout<<"st_no="<<inform.st_no <<endl;
    cout<<"l_name="<<inform.l_name <<endl;

```



```

cout<<"f_name="<<inform.f_name <<endl;
cout<<"m_name="<<inform.m_name<<endl;
cout<<"recordbook="<<inform.recordbook<<endl;
cout<<"tel="<<inform.tel <<endl<<endl;
return 0;
}

```

Дар ин намууди барнома, ки функцияро аз навъи сохтор муаррифӣ кардем ва бо ёрии оператори бахшиш ба аъзоҳои сохтор майдонҳои тағйирёбандаи `inf` бахшида шудааст. Дар ин ҳолат натиҷаи барнома шакли зеринро мегирад:

```

s_no=2
f_name=Habib
m_name= Kurbonovich
recordbook=124
tel=93-333-33-33
s_no=1
l_name=Saidov
f_name=Umar
m_name= Kurbonovich
recordbook=4224
tel=918-88-88-88

```

Ҳангоми равои кардани сохтор ба функция ва даъвати функция бояд навъи сохторҳо ҳам дар функцияи даъваткунанда ва ҳам дар функцияи даъватшаванда яқхела бошад. Дар сурати риоя накардани ин қонуният, барнома ба хатоғӣ рӯбарӯ мешавад. Новобаста аз он ки ду сохтори дорои аъзоҳои яқхела бошанд. Ба барномаи зерин диққат диҳед:

```

#include <iostream>
#include <stdio.h>
using namespace std;
/* Муайян кардани навъи сохтор */
struct Simple_Struct {
    int a, b;
    char ch;
};
/* Муайян намудани сохтор ба сохтори пешина монанд, вале бо дигар ном. */
struct Simple_Struct {
    int a, b;
    char ch;
};
void f1(struct struct_type2 parm);

int main()
{
    struct struct_type arg;
    arg.a = 1000;
    f1(arg); /* Навъи номувофиқ */
    return 0;
}

```

```

}
void f1(struct struct_type2 parm)
{
    printf("%d", parm.a);
}

```

Навъи додаҳое, ки ба воситаи корбар муаррифӣ мегарданд (user-defined data types)

Дар забони C++/C истифодабарандагон (корбарон) имконият доранд, ки номи нав барои навъи додаҳо, яъне ба ягон навъи дода эк-вивалент муаррифӣ карда истифода баранд. Ин амал ба ёрии калимаи калидии typedef анҷом меёбад. Агар ба мантиқӣ барнома нигарем, навъи нав пайдо нагардида, балки ба ягон навъ номи нав дода мешавад. Вақте ки номи нав пайдо шуд, метавонем тағйирёбандаҳои лозимӣ, ҳатто массивҳо ва сохторҳои низ аз он навъ муаррифӣ кунем. Наҳви оператори typedef ба намуди зерин аст:

```
typedef type new_type
```

Дар ин ҷо type навъи дода буда, метавонад дилхоҳ навъи стандартӣ бошад, ба монанди int, float, double, char ва ғайра. Масалан, барои навъи int метавон ба таври зерин номи нав гузошт, яъне

```
typedef int butun;
```

Дар ин ҷо барои навъи int номи нав butun муаррифӣ шудааст. Пас аз ин, дар ҳар ҷойи барнома метавонем, тағйирёбандаҳои навъи бутунро ба номи butun ба таври муаррифӣ кунем:

```
butun a,b;           ё   int a,b;
```

Ба ҳамин монанд метавонем, барои дилхоҳ навъи додаҳо номи нав гузошта истифода барем. **Мисол:**

```

#include <iostream>
#include <stdio.h>
using namespace std;
typedef int butun;
typedef float haqiqi;
int main(){
    butun i,n=10;
    haqiqi S=0,P;
    for (i=0;i<=n;i++)
        S+=i;
    P=S/n;
    cout<<"P="<<P<<endl;
    return 0;
}

```

Дар ин ҷо дида мешавад, ки барои навъи int номи butun ва барои навъи float номи haqiqi муаррифӣ шудааст ва дар барномаи тағйирёбандаҳои i ва n аз навъи butun ва тағйирёбандаҳои S ва P аз навъи haqiqi муаррифӣ шудааст.

Истифодаи typedef ба хусус барои муаррифӣ намудани сохторҳои корро осон мекунад, яъне аз такроршавии калимаи struct озод мешавад. **Мисол:**

```
typedef struct{
    var1;
    var2;
    ....
    Var n;
} new_type;
```

Дар ин ҷо new_type навъи сохторро муаррифӣ мекунад. Аз ин ҷо метавонем аъзоҳо ё тағйирёбандаҳои навъи сохторро бо чунин навъ муаррифӣ кард. Мисол:

```
typedef struct{
    int st_no;
    char f_name[25];
    float balance;
} record;
record new_balance, l_name[25], m_name[25];
```

Дар ин ҷо record аз навъи сохтор муаррифӣ шудааст. Тағйирёбандаҳои new_balance, l_name[25] ва m_name[25] тағйирёбандаҳои сохторӣ буда, аз навъи record муаррифӣ гардидаанд.

Сохтори додаҳо ва ишорагарҳо

Ба монанди тағйирёбандаҳои дигар барои дастрас шудан ба як сохтор аз оператори амперсанд «&» истифода бурда мешавад. Маса-лан, бигзор тағйирёбандаи variable аз навъи сохтор бошад, ифодаи &variable суроғи ибтидоии онро нишон медиҳад. Тағйирёбандаҳои ишорагарро метавонем ба намуди зерин муаррифӣ кунем :

type*ptvariable

Дар ин ҷо type навъи додае аз таркиби сохтор буда, ptvariable муайянкунандаи номи ишорагар мебошад. Мисол:

```
typedef struct{
    int st_no;
    char l_name[30];
    char f_name[30];
    char m_name[30];
    float balance;
} record;
record cust,*ptcust;
```

Дар ин ҷо cust тағйирёбандае аз навъи record буда, ptcust бошад, тағйирёбандаи ишорагар аст, ки дар худ суроғи сохторро нигоҳ медорад. ptcust=&cust;

Ба ғайр аз ин, тағйирёбандаҳои навъи ишорагар метавонанд ба сурати зерин муаррифӣ гарданд.

```
struct {
    member1;
    member2
    ...
    member n
} var,*ptvar;
```

Дар ин ҷо `var` тағйирёбандаи навъи сохтор буда, `*ptvar` бошад, ишорагаре ба сохтор мебошад.

Барои дастрасшавӣ ба аъзои сохтор аз оператори «-» истифода бурда мешавад. Барои дастрасшавӣ ба элементи як сохторе, ки худаш аъзои сохтор аст, оператори мазкур ба оператори дот якҷоя истифода бурда мешавад.

```
ptvar->member.submember;
```

Агар элементи сохтор массив бошад, барои дастрас шудан ба элементҳои массив номи массив ва зернависи массив истифода бурда мешавад, яъне

```
ptvar->member[expression];
```

Мисол:

```
typedef struct{
    int st_no;
    char l_name[30];
    char f_name[30];
    char m_name[30];
    float balance;
} student, *ps=& student;
```

Тавре ки дар ин ҷо дида мешавад, ба тағйирёбандаи ишорагари `ps` суроғаи аввалаи тағйирёбандаи `student`, ки аз навъи сохтор аст, ҳамчун аъзои аввала нисбат дода шудааст, яъне тағйирёбандаи `ps` барои тағйирёбандаи `student` ишорагар мебошад. Барои дастрасшавӣ ба рақами тартибии ягон донишҷӯ се роҳ мавҷуд аст:

- 1) `student.st_no`;
- 2) `ps->st_no`;
- 3) `(*ps).st_no`;

Дар ифодаи охири навиштани қавс ҳатмист, зеро оператори дот аз оператори «*» пурқудраттар аст, дар сурати нагузоштани қавс, шояд компилятор хатоғидиҳад ва барнома кор накунад. Ба ҳамин монанд ба аъзоҳои дигар дастрас шудан мумкин аст. Ҳамин тавр, ба номи донишҷӯи `n`-ум дастрас шудан аз ифодаҳои зерин истифода кардан мумкин аст.

```
student.f_name[n]   ps->f_name[2],   (*ps).f_name[2]
*(student.f_name+2)ps->(name+2)     *((*ps).name+2)
```

Ишорагарҳо ба сифати элементи сохтор

Як тағйирёбандаи ишорагар метавонад, элементи як сохтор бошад. Масалан, метавонем насаб, ном ва номи падари як нафар донишҷӯро ба сурати сохтор ба намуди зерин эълон кунем.

```
struct names{
    char *l_name;
    char *f_name;
    char *m_name;
}
```

Дар ин ҷо `l_name`, `f_name` ва `m_name` се тағйирёбандаи ишорагари рамзӣҳастанд, ки мувофиқан насаб, ном ва номи падар як донишҷӯро нишон медиҳанд.

Ҳамин тариқ аъзои як сохтор метавонад як ишорагар ё суроғи сохтори дигарро дар худ нигоҳ дошта бошад ва ё суроғи сохторро аз навъи худаш ногаҳдорӣ кунад. Дар ҳолати умумӣ ин сохтор намуди зеринро дорад.

```
struct rec
{
    member1
    member2
    ....
    struct rec & f_name;
};
```

Дар ин ҷо `f_name` тағйирёбандае аз навъи ишорагар мебошад, ки дар худ тағйирёбандаи дигарро аз навъи сохтор дар худ бо форми `rec` дар он қарор дод. Яке аз қорбурдҳои равиши мазкур барои сохтани сохторҳои бо номи `linked list`-иттифоқи рӯйхатҳо қорбурди зиёд доранд.

Иттиҳодҳо (Unions)

Иттиҳод (`union`) ҷое аз ҳофиза аст, ки дар он якчанд тағйирёбанда нигоҳ дошта мешавад. Навъи тағйирёбандаҳои метавонанд якхела ё гуногун бошад. Дар воқеъ иттиҳод тағйирёбандаест, ки имконияти захира қардани тағйирёбандаҳои гуногунро дар ҷойи муштараки ҳофиза фароҳам меорад.

Наҳви муаррификунии иттиҳодҳо монанди сохторҳо мебошад, вале бартарияташ дар он аст, ки сохторҳо барои ҳар элементаш як ҷойи ҳофизаро банд мекунад. Иттиҳодҳо бошанд, барои ҳамаи аъзоҳояш танҳо як ҷойе аз ҳофизаро банд мекунад. Ин навъ сохтори додаҳо дар истифодабарихо хело муфид аст, зеро дар он элементҳои чандгонаи навъҳои гуногун ҳастанд ва дар ҳар замон метавонем, ки ба ҳар як аъзо қимате бахшем.

Тарзи эълон намудани иттиҳодҳо шакли зеринро дорад:

```
union номи_иттиҳод
{
    Навъи_тағйирёбандаи1 номи тағйирёбандаи1;
    Навъи_тағйирёбандаи2 номи тағйирёбандаи2;
    ....
    Навъи_тағйирёбандаиN номи тағйирёбандаиN;
} Рӯйхати экземплярҳо;
```

Дар ин ҷо `union` калимаи калидӣ буда, барои муаррификунии иттиҳодҳо истифода бурда мешавад. **Мисол:**

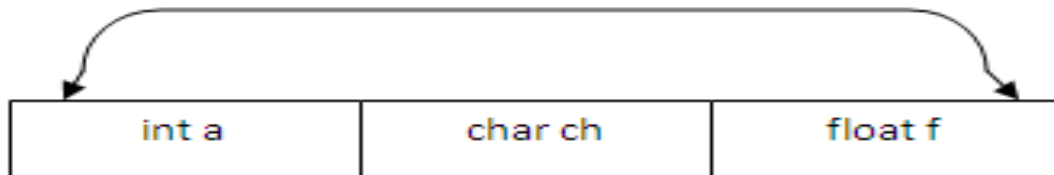
```
union union_type{
    int a1;
    char ch;
    float f;
};
```

Ин эълонкунӣ монанди сохторҳо сабаби муаррифи намудани тағйирёбанда намегардад, балки таърифи як намуди дода аст. Барои муаррификунии тағйирёбандаҳои навъи `union` пеш аз номи тағйирёбанда калимаи калидии `union` навишта мешавад. Наҳви муаррификунии тағйирёбандаҳои навъи `union` монанди тағйирёбандаҳои навъи `struct` мебошад.

Мисол:

```
union union_type{
    int a;
    char ch;
    float f;
} b;
union union_type{
    int a;
    char ch;
    float f;
};
union union_type b;
```

Дар ин ҷо тағйирёбандаи a, ch ва f яке аз навъи int, дуввум аз навъи char ва саввумӣ аз навъи float мебошад. Ҳарчанд ки ин тағйир-ёбандаҳои дар хотира микдорҳои гуногунро банд мекунад, вале су-роғаи онҳо дар якҷост.



Тавре ки аз расм дида мешавад, барои се тағйирёбандаи гуногунавъ як ҷойи хотира банд карда шудааст. Вақте ки union муаррифӣ шуд, компилятор ба таври автоматӣ як тағйирёбандае, ки бузургтарин элементи union-ро дар худ нигоҳ медорад, эҷод мекунад. Ба таври умумӣ, нахви муаррифӣ намудани тағйирёбандаҳои навъи сохтор намуди зеринро дорад.

```
storage_class union union_type var1,var2,..., varN;
```

Дар ин ҷо storage_class як доимии ихтиёрӣ мебошад, ки агар онро истифода набарем, ҳофизаи мавриди назар аз навъи автоматӣ мешавад.

Тавре ки баён кардем, тағйирёбандаҳои навъи union пас аз муаррифӣкунии union, яъне пас аз қавси системавӣ навишта мешаванд ва ё дар алоҳидагӣ муаррифӣ карда мешавад. **Мисол:**

```
union student {
    char name[12];
    int st_no;
} a,b;
```

Дар ин ҷо ду тағйирёбанда аз навъи union-е, ки ба номи student муаррифӣ гардидаанд, ҳар замон метавонанд муаррифӣгарӣ сатри 12-рамза ё адади бутун бошанд. Дар ин ҷо айён аст, ки тағйирёбандаи name нисбат ба тағйирёбандаи st_no ба ҳофизаи зиёд ниёз дорад. Дар натиҷа компилятор барои ҳар як тағйирёбанда ба микдори тағйирёбандаи бузургтар минтақаи ҳофизаро ҷудо мекунад. Мисол:

```
#include <iostream>
#include <stdio.h>
using namespace std;
union adad{
    unsigned short int a;
    int b;
} x;
void print ( adad z){
    cout<<"a="<<z.a <<endl;
```

```

        cout<<"b="<<z.b <<endl<<endl;
    }
    int main(){
        adad y;
        x.b =35;
        print(x);
        y.b =3.5;
        print(y) ;
        return 0;
    }

```

Дар ин барнома тавре дида мешавад, иттиҳоде ба номи adad муаррифӣ гардидааст, ки дорои ду аъзо мебошад, ки якеаш аз навъи unsigned short int буда, дигараш аз навъи int мебошад. Баъдан функсияе бо номи print муаррифӣ шудааст, ки аргументи он аз навъи иттиҳод мебошад, сипас, дар функсияи аслии барои тағйирёбандаи навъи int яъне b қимате бахшида шудааст. Ҳангоми хоричкунӣ, яъне даъвати функсияи print, дар саҳифаи тасвир қимати тағйирёбандаи unsigned short int низ ба он баробар ба чоп дода мешавад.

Як union метавонад узви як сохтор бошад ва инчунин як сохтор низ метавонад узви union бошад. **Мисол:**

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    union student {
        int st_no;
        char jender;
    };
    struct name{
        char l_name[25];
        char f_name[25];
        char m_name[25];
        union student kurs;
    } ;
    static name don[]={
        {"Saidov", "Umar", "Kurbonovich", 65},
        {"Aliev", "Vali", "Saidovich", 'D'}
    };
    cout<<"firstName="<<don[0].l_name<<endl ;
    cout<<"name="<<don[0].f_name <<endl;
    cout<<"LastName="<<don[0].m_name <<endl ;
    cout<<"kurs="<<don[0].kurs.st_no<<endl ;
    cout<<don[0].kurs.jender<<endl<<endl;
    cout<<"firstName="<<don[1].l_name<<endl ;
    cout<<"name="<<don[1].f_name <<endl;
    cout<<"LastName="<<don[1].m_name <<endl ;
    cout<<"kurs="<<don[1].kurs.st_no<<endl ;
    cout<<don[1].kurs.jender<<endl;
    return 0;
}

```

Дар ин мисол узви аввалаи union адади бутун буда, узви дигараш тағйирёбандаи рамзӣ мебошад. Унсурҳои сохтор бошанд, мувофиқан фамилия, ном ва насаби донишҷӯро ифода мекунанд. Ба ғайр аз ин, унсури дигари сохтор union мебошад. Дар функсияи асли ба ҳар як унсури сохтор миқдоре бахшида мешавад. Аз ин ҷо мулоҳиза карда мешавад, ки барои дастрас шудан ба қимате, ки ба унсури сохтор, ки аз навъи union буд, бо ёрии операторҳои зерин амалӣ мешавад.

```
don[0].kurs.st_no;
don[0].kurs.jender;
```

Дар ин ҷо аввал ба узви унсури навъи сохтор адади бутуни ба 65 баробар бахшида шудааст. Тавре ки гуфтем, дар union ҳофиза муштарақ аст, бинобар ин, қимати don[0].kurs.st_no=65 ва қимати don[0].kurs.jender=A ба ҷоп дода мешавд, яъне ASCII-коди A=65 мебошад, баъдан ба узви union сим-воли 'D' бахшида шудааст. Дар ин ҷо низ қимати don[0].kurs.st_no=68 ва қимати don[0].kurs.jender=D баробар ба ҷоп дода мешавад. Натиҷаи барнома ба сурати зерин аст:

```
firstName=Saidov
name=Umar
LastName= Kurbonovich
A
firstName=Aliev
name=Vali
LastName= saidovich
D
```

Додаҳои навъи шуморишӣ

Дар зери мафҳуми додаҳои навъи шуморишӣ дастаи додаҳои бутуни доимӣ фаҳмида мешаванд, ки ҳар яке аз онҳо барои муайянқунии қимати тағйирёбандаҳои навъи шуморишӣ истифода бурда мешаванд. Додаҳои навъи мазкур монанди додаҳои навъи сохтор ва ё union мебошанд.

Пеш аз додаҳои навъи шуморишӣ ҳангоми муаррификунӣ калимаи калидии enum навишта мешавад. Наҳви умумии муаррификунии додаҳои навъи мазкур намуди зеринро дорад.

```
enum constnum { const1, const2, ...,constN};
```

Дар ин ҷо enum калимаи калиди буда, consenum номи ихтиёри мебошад, ки барои додаҳои навъи шуморишӣ истифода шудааст. Пас аз муаррификунии додаҳои навъи шуморишӣ метавонем дар дигар ҷойи барнома додаҳои мувофиқ ба ин навъро ба сурати зерин муар-рифӣ кард:

```
constenum var1,var2,....,varN;
```

Дар ин ҷо onsenum номи навъи шуморишӣ буда, var1,var2,....,varN тағйирёбандаҳои навъи шуморишӣ мебошанд. Ба ғайр аз ин, метавонем додаҳои навъи шумориширо ба сурати зерин муаррифӣ кард:

```
enum constenum { const1, const2, ...,constN}, var1,var2,....,varN;
```

Ҳар як унсури додаҳои навъи шуморишӣ дар ҳолати умумӣ дорои қимати бутун вобаста аз мавқеъашон мебошанд. Дар вақти муаррификунӣ метавонем

ба додаҳои навъи мазкур қимати аввала бахшем. Амали мазкур пас аз рамзи навъи шуморишӣ аломати «=» гузошта иҷро карда мешавад. Мисол:

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main() {
    enum color{red,blue, black,green,magenta,white,yellow,cyan};
    cout<<red<<"\t"<<blue<<"\t"<<black<<"\t"<<green<<"\t"<<magenta<<"
\t"<<white<<"\t"<<yellow<<"\t"<<cyan<<endl;
    return 0;
}
```

Қимати ин рамзҳо ба сурати зерин аст:

0	1	2	3	4	5	6	7
---	---	---	---	---	---	---	---

Яъне дар ин ҷо: red=0; blue=1; black=2; green=3; magenta=4; white=5; yellow=6 ва cyan=7 мебошад.

Агар мо бо ягонтои ин додаҳо қимате ихтисос диҳем, он гоҳ аз он сар карда, рақамгузори карда мешаванд.

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    enum color{red,blue,
black=10,green,magenta,white=20,yellow,cyan};
    cout<<red<<"\t"<<blue<<"\t"<<black<<"\t"<<green<<"\t"<<magenta<<"
\t"<<white<<"\t"<<yellow<<"\t"<<cyan<<endl;
    return 0;
}
```

Натиҷаи ин барнома ба сурати зерин хоҳад буд.

0	1	10	11	12	20	21	22
---	---	----	----	----	----	----	----

Миқдори ин рамзҳо ба сурати зерин аст: red =0, blue=1, black=10, green=11, magenta=12, white=20, yellow=21 ва cyan=22 мебошад.

Саволҳо барои мустаҳкамкунӣ

1. Сабаби аз сохторҳо истифода бурдан дар чист?
2. Чӣ тавр як сохтор муаррифӣ карда мешавад?
3. Оё ҳангоми муаррифӣ намудани як сохтор метавон ба он қимати аввала бахшид?
4. Оё як сохтор метавон дар дохили сохтори дигар муаррифӣ гардад?
5. Ишорагар ба сохтор чист ва чӣ тавр он муаррифӣ карда мешавад?
6. Сабаби аз сохторҳо истифода бурдан дар чист?
7. Чӣ тавр ба сохторҳо ҳангоми эълонкунӣ қимати аввала бахшида мешавад?
8. Оё мумкин аст, ки массивҳо аз навъи сохтор муаррифи шаванд?
9. Оё метавон ба сифати аргументи функсия сохтореро истифода кард?
10. Майдонҳо ё аъзоҳои сохтор оё метавонанд ишорагар бошанд?
11. Иттиҳод чист ва он аз сохторҳо чӣ фарқ дорад?

12. Додаҳои шуморишӣ чӣ гуна додаанд?

Мисолҳои барои кори мустақилона

1. Алӣ аз бозори яклухт миқдори маҳсулотро бо нархҳои гуногун харида, дар мағозаи назди хонааш аз рӯи шартҳои зерин мефурӯшад:

Агар нархи маҳсулот аз 100 сомонӣ кам бошад, дар болояш 8% и нархашро илова мекунад ва агар нархи маҳсулот аз 100 сомонӣ то 1000 сомонӣ бошад, дар болояш 12% и нархашро илова мекунад ва агар маҳсулот аз 1000 сомонӣ зиёд бошад, дар болояш 18% нархашро илова мекунад. Талаб карда мешавад, ки барномае нависед, ки нархҳои илованамуда ва фоидаи ба даст меовардаи Алиро ба намуди ҳисобот (ҷадвал) пешниҳод кунад.

№	Номи маҳсулот	Нархи харид	Фоизи иловашуда	Нархи фурӯш	Фоида
---	---------------	-------------	-----------------	-------------	-------

2. Донишҷӯ дар семестри тобистона аз 5 фанни таълимӣ имтиҳон супорид. Агар ҳамаи баҳоҳои гирифтааш 4 ва 5 бошанд, аз тарафи компани Tcell соҳиби 600 дақиқаи бепул дар телефонаш мегардад. Гурӯҳе иборат аз 25 нафар донишҷӯ имтиҳон супориданд. Барномае нависед муайян кунад, ки чанд нафар донишҷӯ ба мукофот сазовор мегарданд. Ному насаби онҳоро бо баҳоҳои фанҳояшон ба намуди ҷадвал бароред.

3. Рӯйхати муштариёни бонк дода шудаанд: фамилия, миқдори пасандоз, шарт ё фоизи зиёдшавӣ. Барномае нависед, ки амалиёти зеринро муайян кунад ва ба саҳифаи тасвир барорад:

- ✓ Андози солони муштариён дар охири сол;
- ✓ Муштариёне, ки пасандози баробар доранд;

4. Маълумоти донишҷӯёни курсе ба сурати: фамилия, ном, номи падар ва зодрӯз дода шудаанд. Барномае нависед, то муайян кунад, ки оё дар гурӯҳ ягон нафар донишҷӯ дар рӯзи додашуда зодрӯз дорад ё не. Агар дошта бошад, маълумотҳои онро бароред ва дар ҳолати акс пайгоми муносибе чоп кунад.

5. Маълумоти мағозаи компютерфурӯшӣ ба намуди: марка, нарх, соли барориш ва кишвари истеҳсолшуда дода шудаанд. Барномае нависед, ки маркаи аз ҳама гарантаринро бо маълумотҳояш барорад.

6. Таърихи рӯз ба намуди сол, моҳ ва рӯз дода шудааст. Барномае нависед, ки бо дохил кардани таърихи рӯз рӯзи баъдинаро барорад. (рӯз, моҳ ва сол алоҳида алоҳида дода мешаванд.)

7. Рӯйхати автомобилҳое, ки дар истгоҳи нақлиёти қарор доранд, дода шудааст: марка, рақам, соли барориш ва миқдори роҳи тайнамудаи мошин ва ному насаби ронанда. Барномае нависед, ки автомобилҳое, ки роҳи баробар тай карданд, мурағаб карда барорад.

8. Дар ҷадвале маълумот дар бораи ҷавонони ба хизмати ҳарбӣ даъватшаванда ба сурати: ному насаб, дарозии қад ва вазн дода шудааст.

Мувофиқи фармони вазири мудофия ҷавононро, вазнашон аз 45кг кам, баландии қадашон аз 160 см кӯтоҳ набуда, синну солашон дар фосилаи 18-27 ҷойгир аст, ба хизмат даъват карда мешаванд.

Барномае нависед, ки рӯйхати даъватшавандагонро, ки ба талаботи хизмати ҳарбӣ ҷавобгӯянд, мурағаб намуда, ба сурати ҳисобот барорад.

9. Чадвали ҳаракати поездҳо дода шудааст, ки аз маълумоти зерин иборат мебошад: Вақти ҳаракат (соати ҳаракат) ва масири ҳаракат.

Барномае нависед, ки:

- ✓ Бо дохил кардани масири ҳаракат соати ҳаракатро муайян кунад;
- ✓ Бо дохил кардани соат масири ҳаракатро муайян кунад.

Боби 10. Файлҳо

Муқаддима

То ҳол мо барномаҳое, ки навиштем ва истифода кардем, ҳамаи додаҳоро аз клавиатура дохил карда шуда, дар тағйирёбандаҳо, массивҳо ва сохторҳо захира мекардем. Додаҳое, ки ҳаҷми зиёд доштанд, дар массивҳо ба хусус аз навъи сохторҳо захир мешуданд. Масалан, иттилооте комиле, ки вобаста ба 100 нафар донишҷӯ аст, дар массиве аз навъи сохтор ба дарозии 100 захира карда мешуд. Ин амал хело хуб аст, аммо вақте ки барномаро иҷро карданӣ шавем, ба мо лозим мешавад, ки додаҳои ҳамаи донишҷӯёнро ба барнома аз саҳифаи вурудӣ дохил кунем. Додаҳои воридкардашуда дар массиве аз навъи сохтор захира мешуданд ва пардозии лозимӣ болои онҳо гузаронида мешуд. Массив дар ин сурат дар RAM захира мегардад, бо поён расидани кори барнома додаҳо низ аз RAM нест мегарданд. Дафъаи дигар вақте ки барномаро иҷро карданӣ шавем, боз зарурати аз сари нав дохил намудани додаҳо пеш меояд. Ин корро шояд барои 100 нафар карда тавонем, вале агар теъдоди донишҷӯён ба 1000 нафар баробар бошад, иҷрои амали мазкур хеле сангин ва дилгиркунанда гардида, вақти зиёде сарф мешавад.

Агар додаҳои воридкардашударо дар диск захира намоем, ҳар маротиба аз сари нав дохил кардани онҳо душворӣ пеш намеорад. Додаҳо дар диск ба намуди файл захира карда мешаванд, яъне ҳамон тавре, ки барои нигоҳ доштани додаҳо дар ҳофизаи RAM аз массивҳо истифода мешуд, барои нигоҳ доштани додаҳо дар диск аз файлҳо истифода карда мешавад. Ҳар файл дорои ном аст, ва истифода-баранда метавонад файло ба он ном дастбӣ кунад. Барои номгу-зори файлҳо аз қонуни номгузори идентификаторҳо истифода бурда мешавад. Номи файл дорои ду қисмат мебошад, ки бо нукта аз ҳамдигар ҷудо карда мешаванд. Қисмати аввалро номи файл ва қисмати дуввум, яъне қисмати баъди нуктаро пасванди файл меноманд. Пасванди файл дорои се ҳарф ё чор ҳарф буда, дар кадом барнома навиштани онро нишон медиҳад. Мисол Ali.txt, Vali.bat, said.docs ва ғайра.

Барои захира кардани додаҳо дар диск барнома бояд файло кушода иттилооти аз саҳифаи калидӣ хондашударо дар он захира кунад. Файле, ки барнома барои нигоҳ доштани иттилооти воридкардашуда мекушоад, файли хуруҷӣ номида мешавад.

Додаҳое, ки бо воситаи барнома дар файл захира мешаванд, метавонанд ба воситаи барномаи дигар ё ҳамон барнома аз файл хонда шуда, мавриди истифода қарор гиранд. Файле вурудӣ номида мешавад, агар танҳо барои хондани иттилоот кушода шавад. Файлро, ки дар он ҳам амали хондан ва ҳам навиштани додаҳо сурат мегирад файли вурудӣ-хуруҷӣ ном дорад.

Дар забони C++ барои амали дохилкунӣ ва хориҷкуниро ба файлҳо анҷом додан метавон аз усули C ва ё усули C++ истифода кард. Дар забони C/C++ нисбат ба забонҳои дигар функцияҳои зиёде вучуд доранд, ки амалиёти I/O-ро иҷро мекунанд. Баръакси дигар забонҳо, забони C/C++ байни дастбӣ ба намуди мустаким ва тартибӣ фарқ намегузорад. Аслан дар забони C/C++ ду намуди файлҳои додаҳо мавҷуд аст:

- 1) Файли додаҳои ба ҷараён вобаста (stream oriented)

2) Файли додаҳои ба система вобаста (system oriented).

Кор бо файлҳои додаҳои ба чараён вобаста назар ба файлҳои додаҳои ба система вобаста осонтар аст, аз ин рӯ, навъи стандартӣ ё чараёнгарро дар ин боб мавриди баҳс қарор медиҳем.

Файлҳои додаҳои чараёнгар низ ба ду гурӯҳ ҷудо мешаванд:

Гурӯҳи аввал танҳо аз рамзҳо таркиб ёфтаанд, яъне рамз метавонад намуди гуногуни додаҳо бошад, ба монанди ададҳо ва сатрҳо. Барои навиштани додаҳо ва ё хондани додаҳо дар ин навъ файлҳо аз функсияҳои монанди `printf()` ва `scanf()` истифода бурда мешавад. Файли додаҳои ин навъ, файлҳои додаҳои қолабдор номида мешавад.

Гурӯҳи дигари файлҳои чараёнгар файлҳои додаҳои беқолаб номида шуда, додаҳои ин навъ файлҳо сохторҳо ва массивҳо мебошанд. Барои пардозии ин гуна файлҳо боз функсияҳои дигаре низ вучуд доранд. Ин функсияҳо имконият медиҳанд, ки додаҳо як-як ё якбора навишта ва хонда шаванд.

Тавре ки гуфтем, кор бо файлҳои системавӣ назар ба файлҳои чараёнгар душвортар мебошад, чунки файлҳои системавӣ вобаста ба системаи омилӣ мебошанд ва инчунин барои пардозии онҳо функсияҳои махсус лозим аст.

Додаҳо аслан бо яке аз чор усули дар поён овардашуда метавонанд дар файлҳо захира шаванд ва ё аз он хонда шаванд.

1. Ҳамчун як рамз:

Ҳар маротиба як рамз хонда ё навишта мешавад, яъне додаҳо як рамз навишта ё хонда мешаванд. Ба монанди он ки дар гузашта мушоҳида кардем, ки як рамз бо ёрии функсияҳои `getchar()`, `putchar` хонда ё навишта мешуд. Ин функсияҳо аслан дар услуби C ҳастанд, барои C++ бошад, функсияи `get()` ва `getline()` ин амалро иҷро мекунанд.

2. Ҳамчун як сатр

Дар ин равиш додаҳо ба намуди маҷмӯи рамзҳо яъне як сатр хонда ё навишта мешаванд. Барои иҷрои ин амалҳо функсияи `gets()` (барои хондани як сатр) ва функсияи `puts()` (барои навиштани як сатр) истифода бурда мешавад.

Додаҳо ҳангоми навиштан бо як формати махсусе навишта шуда, дар вақти хондан бо он формат хонда мешаванд.

Барои иҷрои ин амал аз функсияҳои `scanf()` ва `printf()`, ки шумо бо онҳо шинос ҳастед, истифода бурда мешаванд.

3. Ҳамчун сохтор ё блок.

Дар ин навъ додаҳо ба намуди як блок ё сохтор дар файлҳо навишта ё хонда мешаванд. Блок гуфта, маҷмӯи ифодаҳо (ё операторҳои) бо ҳамдигар аз ҷиҳати мантиқӣ алоқамандро меноманд, ки дар дохили қавсҳои системавӣ «{...}» навишта мешаванд.

Ҳангоми навиштан ё хондани додаҳо дар файлҳо системаи омилӣ метавонад, таъсир расонад. Функсияҳои I/O –и стандартӣ ANSI дар забони C/C++ метавонад файлҳоро ба ду намуд: матнӣ ва бинарӣ дастгирӣ намоянд.

Файлҳо ва чараёнҳо (Files and streams)

Пеш аз он ки мо дар бораи системаи I/O сухан гӯем, бояд фарқи байни чараён ва файло бидонем. Забони C /C++ барои барномасоз як василаеро ба вучуд меорад, ки нисбат ба дастгоҳ ҷудо аст, яъне як ҷудоии байни барномасоз ва дастгоҳ. Ин ҷудоисозии чараён номида мешавад. Дастгоҳ аслан бингарем, ҳамон як файл аст ва чараён воситаест байни он ва барнома ба монанди пул (мост) алоқа барқарор мекунад.

Чараёнҳо

Яке аз хусусиятҳои забони C/C++ дар он аст, ки байни дастгоҳҳо фарқ намегузорад, яъне барои C/C++ фарқ надорад, ки файл дар диски мантиқӣ ё ягон барандагони иттилооти дигар дода шудааст. Ҳама вақт амали I/O ба воситаи як чараён, ки ба файлҳо вобаста аст, амалӣ мегардад. Агар як чараёнро аз рӯи байтҳо ба назар гирем, намуди зеринро дорад.

0	1	2	3	...	$n - 1$	EOF
---	---	---	---	-----	---------	-----

Тавре ки дар мавзӯи сатрҳо гуфта будем, ҳар сатр ба як рамзи NULL ба анҷом мерасад. Дар ин ҷо метавон гуфт, ки ҳар файл бо як рамзи EOF ба анҷом мерасад. Калимаи EOF низ шакли мухтасари End Of File буда, доими китобхонаи `stdio.h` ба ҳисоб меравад. Вақте ки як файл барои навиштан ё хондан кушода мешавад, дар ин ҳолат як чараён низ эҷод мегардад. Ба таври дигар гӯем, забони C/C++ додаҳои як файло ба монанди як массиви рамзии якченака дастбӯӣ мекунад. Навиштани додаҳо дар як файл ин навиштани додаҳои як чараён ва хондани додаҳои як файл хондани як чараён мебошад. Пас, барои хондан ва навиштани файл бояд, ки чараёне эҷод гардад. Як чараён ҳангоми муаррификунии як тағйирёбандаи ишорагаре аз навъи сохтори файл ба вучуд меояд. Як барнома метавонад дорои як ё якчанд чараён бошад.

Чараёнҳои стандартӣ (standard streams)

Ҳамаи он амалиёти дохилкунӣ ва хоричкунӣ, ки то ҳоло мавриди баррасӣ ва баҳс қарор дода будем, аз ду стандарт иборат буданд, яъне барои дохилкунӣ стандарти `stdin` (standard input) ва барои хоричкунӣ аз стандарти `stdout` (standard output). Ба сифати чараёнҳои дохилкунӣ ва хоричкунӣ дар забони C++ се чараён истифода бурда мешаванд.

1. `stderr`
2. `stdout`
3. `stdin`

Ин чараёнҳо ба тарминали мо ишора мекунанд ва мувофиқан барои иҷрои амали хабар додан дар ҳолати хатогӣ, хоричкунӣ ва дохилкунӣ истифода бурда мешаванд. Аз рӯи наҳви захираи додаҳо дар забони C/C++ ду стандарт вучуд дорад:

- ✓ стандартӣ матнӣ (`textstream`)
- ✓ бинарӣ (`binarystream`),

яъне дар забони C/C++ додаҳо ба ду намуд: бинарӣ ва дар қолаби ASCII захира мешаванд.

Аз ин намоиш дар файҳои матнӣ адади бутуни кӯтоҳ, ки аз се рамз иборат аст, дар хотира се байтро ишғол мекунад ва адади дуум, ки дорои 7 рамз аст, дар ҳофиза 7 байтро ишғол мекунад. Тавре ки дар мавзӯи сатрҳо медонем, ҳар як рамз як байт ҳисоб карда мешавад. Ин ададҳоро агар дар файлҳои бинори захира кунем, барои адади 234, ки аз навъи бутуни кӯтоҳ аст, 2 байт ва барои адади ҳақиқии 3,14159 ҳамаги 4 байт аз ҳофиза банд карда мешаваду халос.

Кор бо файлҳои матнӣ

Дар забони C++ барои кор кардан бо файлҳои матнӣ ду ҷараён истифода бурда мешавад. Ҷараёни `ifstream`-барои кор кардан бо файл дар речаи хониш ва ҷараёни `ofstream` - барои кор кардан бо файл дар речаи навиштан. Агар хоҳем, ки ҳам дар речаи хониш ва ҳам дар речаи навиштан бо файл кор кунем, он гоҳ лозим меояд, ки аз ҷараёни `fstream` истифода барем. Барои ба ин ҷараёнҳо дастрас шудан лозим аст, ки библиотекаи `<fstream.h>` пайваст карда шавад.

Додаҳо аслан дар файли матнӣ аз рӯи алгоритми зерин навишта мешаванд:

1. Тағйирёбандае аз навъи ҷараёни `ofstream` муаррифӣ карда мешавад.
2. Файл бо ёрии функсияи `open()` кушода мешавад.
3. Маълумотҳои зарурӣ дар файл навишта мешаванд.
4. Файл бо ёрии функсияи `fclose()` пӯшонидани ё маҳкам карда мешавад.

Сохтани як файли матнӣ ва навиштани додаҳо дар он

Барои навиштани додаҳо дар як файли матнӣ аввал тағйирёбандае аз навъи `ofstream` ба сурати зерин муаррифӣ карда мешавад:

```
ofstream ofs;
```

Дар ин ҷо `ofs` номи ҷараён мебошад. Барои номгузори ҷараёнҳо низ аз қонуни номгузори идентификаторҳо истифода мебарем.

Барои навиштани додаҳо дар файл бояд, ки аввал файл кушода шавад. Ин амал бо ёрии функсияи `open()` ба сурати зерин иҷро карда мешавад:

```
ofs.open("D:\\File_name", mode)
```

Дар ин ҷо `ofs` тағйирёбандае аз навъи ҷараёни `ofstream`, `open()` функсияе, ки бояд файло боз кунад ва ё файли нав эҷод кунад, `Filename` –номи файл (барои номгузорӣ ба файлҳо низ аз қонуни номгузори идентификаторҳо истифода мебарем.) ва `mode` бошад, речаи кушодани файло ифода мекунад, яъне он мушаххас мекунад, ки файл бо чи мақсад кушода шудааст.

Речаҳои, ки бояд бо ёрии яке аз онҳо файли матнӣ кушода шавад, дар ҷадвали зерин оварда шудаанд:

№	Навъи файл	Маъно ё амали иҷро мешудагӣ
1	<code>ios::in</code>	Кушодани файл дар речаи хониш
2	<code>ios::out</code>	Танҳо барои навиштан кушодани файли нав. Агар файли нав бо ягон файли мавҷудбуда ҳамном бошад, файли кӯҳна нест мешавад.
3	<code>ios::app</code>	Файли мавҷудбударо (файли дуй) барои дар охираш илова намудани додаҳо мекушояд. Агар бо номи нишондодашуда файл мавҷуд набо-шад, ба таври автоматӣ эҷод мегардад.

4	ios::trunk	Тоза кардани файл.
5	ios::nocreate	Амали кушодани файло ичро намекунад, агар файл мавчуд набошад.
6	ios::noreplace	Файли мавчудбуда кушода намешавад.

Чадвали 1.

Агар хангоми кушодани файл mode нишон дода нашуда бошад, компилятор ба таври худкура (автоматӣ) барои чараёни ifstream mode-и ios::in ва барои чараёни ofstream mode-и ios::out-ро истифода мебарад. Барои кушодани як файл як чанд роҳ мавчуд аст, ки бо ёрии яке аз онҳо метавон файли лозимиро кушод. Бигзор файле ба номи sample.txt дар диски D мавчуд бошад, пас, барои кушодани он яке аз равиш (роҳ)-ҳои зерин истифода бурда мешаванд:

- 1) ofstream ofs;
ofs.open("D:\\sample.txt", ios::out);
- 2) ofstream ofs;
ofs.open("D:\\sample.txt");
- 3) ofstream ofs ("D:\\sample.txt", ios::out);

Хангоме, ки амали кушодани файл ба таври табиӣ анҷом мепазирад, тағйирёбандаи ofs дорои қимати true мегардад. Дар акси ҳол он ба қимати false баробар мешавад.

Пас аз он ки файл бо муваффақият дар речаи навиштан кушода шуд, бояд додаҳои лозимӣ дар он навишта шаванд. Навишти додаҳо дар файли матнӣ монанди навиштани додаҳо дар саҳифаи тасвир буда, фарқият дар он аст, додаҳо дар саҳифаи тасвир бо ёрии объекти cout анҷом мегирифтанд, вале дар ин ҷо бошад, бо ёрии ҳамон тағйирёбандае, ки аз навъи чараёни ofstream эълон гардидааст. Масалан, барои навиштани як сатр дар файл, оператори зерин истифода бурда мешавад.

```
ofs<<"In Sample File ast!";
```

Амали навиштан адад бошад, намуди зеринро дорад:

```
ofsy<<5;
```

Пас аз навиштани додаҳо дар файл бояд, ки файл баста шавад. Агар файл баста нашавад, пас аз итмомрасии кори барнома он ба таври худкура (автоматӣ) баста мешавад, вале дар ин сурат додаҳои дар чараён буда ба файл навишта намешаванд. Пас, барои нигоҳ доштани иттилооте, ки дар як чараён ҳастанд, ба мо лозим аст, ки бо ёрии функсияи close() файло маҳкам кунем, ин функсия ба сурати зерин истифода бурда мешавад:

```
ofs.close();
```

Акнун, ки ба навиштани додаҳо дар файл шинос шудем, мисолеро дида мебароем.

Мисол. Барномае нависед, ки чадвали зарби ададҳои аз 1 то 10-ро дар файли хориҷ кунад.

```
#include <iostream>
#include <fstream>
using namespace std;
int main(){
```

```

ofstream ofs;
ofs.open("simple.txt");
int i, j;
for(i=1;i<=15;i++)
    for(j=1;j<=15;j++){
        ofs<<i*j<<"\t ";
        cout<<endl;
    }
return 0;
}

```

Хондани додаҳо аз як файли матнӣ

Барои хондани додаҳо аз як файли матнӣ, дар аввал тағйирёбан-дае аз навъи ifstream ба сурати зерин эълон карда мешавад:

```
ifstream ifs;
```

Барои хондани додаҳо аз файл бояд, ки аввал файл кушода шавад. Бо ёрии оператори функсияи open() файли лозимиро мекушояд:

```
ifs.open("D:\\File_name", mode)
```

Дар ин ҷо ifs тағйирёбандае аз навъи чараёни ifstream, open() зерфунксияе, ки бояд файло боз кунад, Filename –номи файл (номи файл метавонад аз пайдарпайии ҳарфҳо, рақамҳо ва ё рамзи хати поён иборат бошад) ва mode бошад, речаи кушодани файло муайян мекунад, яъне он мушаххас мекунад, ки файл ба чи хотир кушода шудааст. Ҳамаи он речаҳое, ки файл бо яке аз онҳо кушода мешавад, дар ҷадвали 1 оварда шудаанд.

Пас аз он ки файл бомуваффақият дар речаи навиштан кушода шуд, бояд додаҳои лозимӣ аз он хонда шаванд. Хондани додаҳо аз файли матнӣ низ монанди хондани додаҳо аз саҳифаи калидӣ буда, фарқият дар он аст, додаҳо аз саҳифаи калидӣ тавассути объекти cin анҷом мегирифтанд, вале дар ин ҷо бо ёрии ҳамон тағйирёбандае, ки аз навъи чараёни ifstream эълон гардида буд, хонда мешавад. Масалан, барои хондани як адади бутун аз файл оператори зерин истифода бурда мешавад.

```
ifs<<a;
```

Агар дар файли мавриди назар ду адад навишта шуда бошад ва дар байни онҳо аломати фосила мавҷуд бошад, оператори ifs<<a рамзи фосиларо ҳамчун охири файл мефаҳмад. Аз ин ҷо саволе ба миён меояд, ки чӣ бояд кард, то ҳамаи додаҳои файло ба даст овард? Барои ҳамаи додаҳои файло ба даст овардан бояд, ки он пас аз ҳар хониш санчида шавад, ки оё интиҳои файл аст ё не. Барои муайянкунии интиҳои файл функсияи eof() ба сурати ifs.eof() истифода бурда мешавад. Ҳангоме ки идоракунӣ(control) ба охири файл мерасад, тағйирёбандаи ifs дорой қимати true мегардад. Барои ҳамаи додаҳои дар файл мавҷуд бударо хондан ин функсия дар дохили ҳалқаи такрорӣ while ё do... while истифода карда мешавад.

Мисол. Дар як файли матнӣ элементҳои массиви навъи бутун навишта шудаанд. Ҳамаи элементҳоро дар саҳифаи тасвир ба чоп диҳед.

```

#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    ifstream ifs;
    int k=0,a;
    ifs.open("simple.txt");
    if (ifs){
        while (!ifs.eof() )
        {
            ifs>>a;
            cout<<a<<" ";
            k++ ;
        }
        ifs.close() ;
        cout<<endl;
        cout<<"k="<<k<<endl;
    }
    else
        cout<<"File not found\n";
    return 0;
}

```

Ҳангоми хондани додаҳо аз файл, метавонем, ки додаҳои навро низ дар файл ҳамроҳ кунем. Барои иҷрои ин амал аз ҳар ду ҷараён ҳам ofstream ва ҳам ifstream истифода мебарем. Дар аввал тавассути тағйирёбандаи ҷараёни ifstream додаҳоро хонда, баъд бо ёрии тағйирёбандаи ҷараёни ofstream додаҳои навро дар файл менависем.

Мисол. Элементҳои массив дар файли матнии simple.txt мавҷуд мебошанд. Барномае нависед, ки ин додаҳоро дар саҳифаи тасвир пешниҳод намуда, сипас, ҳар як элементро як воҳид афзуда, дар охири файл илова кунад.

```

#include <iostream>
#include <fstream>
using namespace std;
int main(){
    ifstream ifs;
    ofstream ofs;
    int k=0,i;
    int *a, b;
    a=new int[100];
    ifs.open("simple.txt",ios::in);
    if (ifs)
    {
        while (!ifs.eof() )
        {
            ifs>>a[k];
            cout<<a[k]<<" ";
            k++ ;
        }
    }
}

```

```

        cout<<endl;
        cout<<"k="<<k<<endl;
        for (i=0;i<=k;i++)
            a[i]=a[i]+1;
        ifs.close() ;
        ofs.open("simple.txt",ios::app);
        ofs<<"\t";
        for (i=0;i<=k;i++)
            ofs<<a[i]<<" ";
        ofs.close() ;
    }
    else
        cout<<"File not found\n";
    return 0;
}

```

Ҳангоми як маротиба иҷро гаштани барнома ҳамаи додаҳои файл дар саҳифаи тасвир пайдо мешаванд ва дар файл низ додаҳои нав навишта мешаванд. Вале ба ғайр аз додаҳои дар файл мавҷудбуда, дар охири додаҳо дар саҳифаи тасвир як ададе ба андозаи навъии `int` низ чоп мешавад. Ин метавонад, ҳангоми якчанд маротиба иҷро намудани барнома ба кори барнома ҳалал ворид кунад. Ба хоҳири он ки ин адад дар саҳифаи тасвир чоп нагардад, ба мо лозим аст, ки оператори охири барномаро ба сурати зерин тағйир диҳем.

```

        for (i=0;i<=k;i++)
            if (i<k-1)
                ofs<<a[i]<<" ";
            else
                ofs<<a[i];

```

Пас аз тағйирдиҳӣ барномаи мо намуди зеринро мегирад.

```

#include <iostream>
#include <fstream>
using namespace std;
int main(){
    ifstream ifs;
    ofstream ofs;
    int k=0,i;
    int *a, b;
    a=new int[100];
    ifs.open("simple.txt",ios::in);
    if (ifs)
    {
        while (!ifs.eof() ){
            ifs>>a[k];
            cout<<a[k]<<" ";
            k++ ;
        }
        cout<<endl;
        cout<<"k="<<k<<endl;
        for (i=0;i<=k;i++)
            a[i]=a[i]+1;

```

```

        ifs.close() ;
        ofs.open("simple.txt",ios::app);
        ofs<<"\t";
        for (i=0;i<k;i++)
            if (i<k-1)
                ofs<<a[i]<<" ";
            else
                ofs<<a[i];
        ofs.close() ;
    }
    else
        cout<<"File not found\n";
    return 0;
}

```

Кор бо файлҳои бинорӣ ё дуӣ

Тавре ки қаблан қайд карда будем, додаҳо дар файлҳои матнӣ ба сурати рамзҳо захира карда мешаванд. Дар файлҳои бинорӣ бошад, додаҳо ба сурати пайдарпайи байтҳо навишта ва ё аз он хонда мешаванд. Барои кор кардан бо файлҳои бинорӣ функсияҳои зиёде истифода бурда мешаванд. Амали сохтан ва навиштани додаҳо дар як файли бинорӣ аз рӯи алгоритми зерин иҷро карда мешавад.

1. Тағйирёбандаи ишорагаре аз навъи FILE муаррифӣ карда мешавад. Дар ин ҷо FILE сохтори файл аст.
2. Бо ёрии функсияи fopen файл кушода мешавад.
3. Додаҳо тавассути функсияи fwrite() дар файл навишта мешаванд.
4. Баъд аз навиштани додаҳо файл тавассути функсияи fclose() баста ё маҳкам карда мешавад.

Барои хондани додаҳо аз як файли бинорӣ бошад, аз алгоритми зерин истифода бурда мешавад:

1. Тағйирёбандаи ишорагаре аз навъи FILE муаррифӣ карда мешавад, Дар ин ҷо FILE сохтори файл аст.
2. Бо ёрии функсияи fopen файл кушода мешавад.
3. Додаҳо тавассути функсияи fread() аз файл то расидан ба поёни файл хонда мешаванд.
4. Баъд аз навиштани додаҳо файл тавассути функсияи fclose() баста ё маҳкам карда мешавад.

Функсияҳои зиёде барои кор кардан бо файлҳои бинорӣ мавҷуданд, ки баъдтар дар ин бора эзоҳ хоҳем дод.

Системаи I/O-и буфердор (buffering I/O system)

Ба ғайр аз функсияҳои мавриди истифода қарор додем, дар забони C++ функсияҳои дигаре барои кор бо файлҳо мавҷуданд, ки онҳоро системаи I/O-и буфердор мегӯянд. Системаи I/O-и забони C++ дар китобхонаи <stdio.h> муаррифӣ карда шудааст. Барои бо он дастрас шудан бояд китобхонаи мазкур

пайваст карда шавад. Функцияҳои муҳимтарини системаи I/O- и буфердор дар ҷадвали зерин оварда шудаанд.

№	Номи функция (function name)	Амали иҷро шаванда (operation)
1	fopen()	Як stream-ро мекушояд.
2	fclose()	Як stream-ро маҳкам мекунад.
3	puts()	Як рамзро ба як stream менависад.
4	gets()	Як рамзро аз як stream мехонад.
5	fseek()	Байти нишондодашуда (мушаххасшуда)-ро дар stream ҷустуҷӯ мекунад.
6	fprintf()	Ба функцияи printf() монанд амалро иҷро мекунад.
7	fscanf()	Амали ба функцияи scanf() монандро иҷро мекунад.
8	feof()	Агар ба интиҳои файл расила бошад, қимати true-ро бармегардонад.
9	ferror()	Дар вақти ба амал омадани хотогӣ қимати true-ро бармегардонад.
10	rewind()	Аз мавқеи охири файл ба аввали файл бармегардад.
11	remove()	Як файдро нест мекунад.
12	freopen()	Як файдро кушода, операторҳои дохилкунӣ ва хоричкуниро ба он вобаста мегардонад

Бояд қайд кард, ки функцияҳои системаи I/O- и буфердор то пур шудани буфер маълумотхоро дар он нигоҳ медоранд. Пас аз пур шудани буфер амалиёти лозимиро болои додаҳои захирашуда анҷом медиҳанд.

Ишорагари файл (File pointer)

Унсуре асосие, ки системаҳои I/O –и буфердорро бо ҳамдигар мепайвандад, ишорагари файл аст. Ишорагари файл тағйирёбандаест, ки ҳолати файл(status File), мавқеи ҷориини файл ё номи файдро ифода мекунад (нишон медиҳад). Ишорагари мазкур дар библиотекаи <stdio.h> муаррифӣ карда шудааст. Наҳви умумии истифодаи ин ишорагар ба сурати зайл аст: FILE *ptvar;

Кушодан ва пӯшидани як файли бинорӣ

Вақте ки бо файлҳои додаҳои ҷараёнгар кор мекунем, бояд пеш аз ҳама, як микдореро аз ҳофиза ҷудо кунем, то ин ки додаҳо дар вақти мубодила дар он ба таври муваққатӣ нигоҳ дошта шаванд. Ин амал дар сурате иҷро мегардад, ки агар ишорагар ба файл муаррифӣ гардида бошад. Наҳви муаррифии ишорагари файл шакли зеринро доارد: FILE *ptvar

Дар ин ҷо FILE як навъ сохтор (structure)-и махсусест, ки барои дар буфер нигоҳ доштан минтақае аз пешбинӣ шудааст. Тағйирё-бандаи ишорагари *ptvar бошад, ба ибтидои он мавқеъ ишора мекунад.

Барои ин ки як файл эҷод ё пардозииш карда шавад, зарур аст, ки он кушода шавад. Амали мазкур номи файдро ба минтақаи буфер вобаста намуда, муайян мекунад, ки файл барои ҷӣ кушода шудааст. (масалан, барои хондан

(read only File) ё барои навиштан (write only File) ва ё инки ҳам барои хондан ва ҳам барои навиштан.

Барои кушодани як файл функсияи fopen() истифода бурда мешавад, ки нахви он ба сурати зерин аст:

Ptvar=fopen(File_name, File_type)

Дар ин ҷо ptvar як тағйирёбандаи ишорагар ба номи FILE мебошад, параметри File_name номи файл ва параметри File_type бошад, сабаби кушодани файло мушаххас мекунад. Функсияи fopen() як чараёнро барои истифода файл эҷод мекунад. Нахви умумии функсияи мазкур намуди зеринро дорад:

FILE *fopen(char *File_name, *mode)

Дар ин ҷо mode –сатрест, ки вазъияти кушодашавии файло нишон медиҳад. Дар забони C/C++ mode ё File_type-ҳои зиёде мавҷуданд, ки муҳимтаринашон дар ҷадвали зерин оварда шудааст:

№	Навъи файл	Маъно ё амали иҷро мешудагӣ
1	“r”, “rt”	Файли додаи мавҷудбударо (файли матнӣ) барои хондан мекушояд.
2	“w”, “wt”	Як файли навро танҳо барои навиштан мекушояд. Агар файли нав бо ягон файли мавҷудбуда ҳамном бошад, файли кӯҳна нест мешавад.
3	“a”, “at”	Як файли мавҷудбударо (файли матнӣ) барои илова намудан мекушояд. Агар бо номи нишондодашуда файл мавҷуд набошад, ба таври автоматӣ сохта мешавад. Иттилооти нав дар охири файл захира мешавад.
4	“r+”, “r+t”	Як файли мавҷудбударо ҳам барои хондан ва ҳам барои навиштан мекушояд.
5	“w+”, “w+t”	Файли наве ҳам барои хондан ва ҳам барои навиштан мекушояд. Агар бо ин ном файл мавҷуд бошад, нест мешавад.
6	“a+”, “a+t”	Файли мавҷудбударо ҳам барои хондан ва ҳам барои илова намудан мекушояд. Агар бо ин ном файл мавҷуд набошад, ба таври автоматӣ файли нав сохта мешавад.
7	“rb”	Як файли бинориро барои хондан мекушояд.
8	“wb”	Як файли бинориро барои навиштан мекушояд.
9	“ab”	Файли бинорро барои илова намудани маълумотҳои нав мекушояд.
10	“r+b”	Як файли бинориро барои хондан ё навиштан мекушояд.
11	“w+b”	Як файли бинориро ҳам барои хондан ва ҳам барои навиштан мекушояд.
12	“a+b”	Як файли бинориро барои хондан ё навиштани мекушояд ва ё аз нав месозад.

Чи хеле ки аз ин ҷадвал мушаххас шуд, як файл метавонад барои хондан ё навиштан ва ё барои хондану навиштан кушода ё эҷод гардад. Инчунин аз ҷадвал айён мегардад, ки файлҳо метавонанд ба намуди матнӣ ё бинорӣ кушода ва ё эҷод гарданд, яъне mode-ҳои бинорӣ аз mode-ҳои матнӣ фарқ доранд.

Масалан, агар мо хоҷем, ки файли матниеро барои навиштан кушоем, оператори зерин кифоягӣ мекунад.

```
fp=fopen("text","r")
```

Дар ин ҷо `fp` ишорагаре бо номи файл аст. Оператори мазкур дар амалия аслан ба намуди зерин истифода бурда мешавад. Мисол:

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main()
{
    FILE *fp;
    if ((fp=fopen("text","w"))==NULL)
    {
        puts("Cannot open File\n");
        exit(1);
    }
    return 0;
}
```

Рабиши мазкур ҳар ҳатогие, ки ҳангоми кушодани файл пеш меояд ошкор менамояд. Аслан ҳатогихо дар ҳолати пур будани диск ва дигар ҳолатҳои монанди ин пеш меоянд. `NULL` бошад, як макросе мебошад, ки дар библиотекаи `<stdio.h>` нигоҳ дошта мешавад. Дар ин ҷо он барои санчиши шарт истифода шудааст.

Тавре ки қаблан баён карда будем, агар функцияи `fopen()`-ро барои хондани як файл истифода барем, ба ин ном файл мавҷуд набошад, файли нав сохта мешавад. Агар хоҷем, ки додаҳои лозимиро дар охири файл нависем, истифодаи `mode`-и “`a`” кифоя аст. Дар ин ҳолат низ бо номи мавриди назар файли мавҷудбуда нест мешавад.

Як чизро набояд фаромӯш кард, ки дар ҳолати нолозим шудани файл бояд он баста шавад, яъне маҳкам гардад. Барои бастан ё маҳкам намудани як файл функцияи `fclose()` даъват карда мешавад. Мисол:

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    FILE *fp;
    if ((fp=fopen("text","w"))==NULL)
    {
        puts("Cannot open File\n");
        exit(1);
        fclose(fp);
    }
    return 0;
}
```

Дар ин ҷо агар файл бомуваффақият кушода нашавад, пайгоми “`Cannot open File\n`” дар саҳифаи тасвир намоиш дода шуда, файл маҳкам карда мешавад. Ин мисолро метавон низ ба намуди зерин навишт.


```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    FILE *fp;
    fp=fopen("text","w");
    if (fp==NULL)
    {
        puts("Cannot open File\n");
        exit(1);
        fclose(fp);
    }
    return 0;
}

```

Тавре ки дар гузашта баён карда будем, забони C/C++ нисбат ба дигар забонҳо функсияҳои зиёдеро барои навиштан ва хондани додаҳо аз файл истифода мебарад. Ҳоло бошад, якчандто аз ин функсияҳоро дида мебароем.

Функсияи `putc()`

Функсияи `putc()` барои навиштани рамзе ба як `stream`, ки пештар тавассути `fopen()` кушода буд, хизмат мекунад. Наҳви умумии ин функсия намуди зеринро дорад.

```
int putc(int chr, FILE *fp)
```

Дар ин ҷо `fp`-ишорагаре ба файл аст, ки онро функсияи `fopen` бармегардонад. Агар амали функсияи мазкур бомуваффақият ба анҷом нарасад, рамзи мавриди назарро бармегардонад, дар акси ҳол қимати `EOF` бармегардонад. `EOF` Тавре ки қаблан гуфта шуд, як макросе аз китобхонаи `<stdio.h>` аст, ки интиҳои файलो ифода мекунад, яъне он шакли тахфифшудаи `End of File` мебошад.

Функсияи `getc()`

Функсияи `getc()` барои хондани як рамз аз файле, ки бо ёрии функсияи `fopen()` кушода шудааст, хизмат мекунад. Наҳви умумии истифодаи ин функсия намуди зеринро дорад:

```
int getc(int chr, FILE *fp)
```

Дар ин ҷо низ `fp` ишорагаре аз навъи `FILE` аст, ки бо ёрии функсияи `fopen()` баргардонида шудааст. Функсияи мазкур як миқдор бутунро бармегардонад ва вақте ки ба охири файл расид, як аломати `EOF`-ро бармегардонад. Барои хондани як файл то охираш метавонад китъаи барномаи зерин кифоят кунад. Мисол:

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    FILE *fp;
    char chr;
    fp=fopen("text","r+t");

```

```

if (fp!=NULL)
{
    chr=getc(fp);
    while(chr!=EOF)
    {
        printf("%c",chr);
        chr=getc(fp);
    }
}
else
{
    puts("Cannot open File\n");
    exit(1);
    fclose(fp);
}
return 0;
}

```

Дар ин мисол барои дар саҳифаи тасвир намоиш додани рамзҳои хондашуда функсияи `printf(«%c»,chr)` истифода бурда шудааст.

Функсияи `feof()`

Тавре ки баён гардид, системаи файли буфердор метавонад бо файлҳои бинорӣ низ амалеро анҷом диҳад. Вақте ки файлҳои бинорӣ кушода мешавад, интиҳои он низ адади бутун ба ҳисоб меравад. Пас саволе ба миён меояд, ки чӣ тавр интиҳои ин навъ файлҳоро муайян карда мешавад? Барои ҳалли ин мушкили функсияи `feof()` истифода бурда мешавад. Функсияи мазкур ҳангоми расидан ба охири файл қимати 1 ва дар акси ҳол, қимати 0-ро бармегардонад. Қитъаи барномаи зерин як файлро то ба интиҳои он расидан мехонад.

```

while (!feof(fp))
    ch=getc(fp);

```

Функсияи мазкурро метавон барои файлҳои матнӣ низ истифода бурд. Ҳоло мисоли дар боло овардашударо тавассути ин функсия дида мебароем.

Мисол:

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    FILE *fp;
    char chr;
    fp=fopen("text","r+t");
    if (fp!=NULL){
        chr=getc(fp);
        while(!feof(fp)){
            printf("%c",chr);
            chr=getc(fp);
        }
    }
    else

```

```

    {
        puts("Cannot open File\n");
        exit(1);
        fclose(fp);
    }
    return 0;
}

```

Функсияи freopen()

Функсияи мазкур файдро барои навиштан ва ё хондан боз намуда, ҷараёнҳои дохилкунӣ ва хоричкуниро ба он пайваст мекунад. Наҳви умумии истифодабарии функсияи мазкур намуди зеринро дорад:

freopen(«file_path»,»type»,»STREAM»).

Дар ин ҷо «file_path» – номи файл ё роҳ ба файл, «type» речаи кушодани файл (ба ҷадвали мавзӯи “Кушодан ва пӯшидани як файли бинорӣ” нигаред) ва «STREAM» – ҷараёни дохилкуни ва хоричкунӣ мебошад. Ҷараёни дохилкунӣ ва хоричкунӣ яке аз стандартҳои stdin, stdout ё stderr мебошад. Мисол ду адади бутун дар файли матнӣ дода шудааст. Барномае нависед, ки суммаи онҳоро ёфта дар ҳамон файл илова кунед.

Варианти якум. Бо истифода аз тобёҳои scanf() ва printf() .

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    long a,b;
    freopen("input.txt","r",stdin);
    freopen("output.txt","w",stdout);
    scanf("%ld%ld",&a,&b);
    printf("%ld",a+b);
    return 0;
}

```

Варианти дуввум. Бо истифода аз объектҳои cin ва cout.

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(){
    long a,b;
    freopen("input.txt","r",stdin);
    freopen("output.txt","w",stdout);
    cin>>a>>b;
    cout<<a+b;
    return 0;
}

```

Истифодаи массивҳои сеченака

То ҳол мо массивҳои якченака ва дученакаро истифода кардем. Массивҳои сеченака низ баъзан барои ҳалли масъалаҳо истифода бурда

мешаванд. Тавре ки медонем, нахви муаррификунии массиви сеченака шакли зеринро дорад:

тип номи массив[андозаи якум][андозаи дуум][андозаи сеюм];

Дар ин ҷо ба сифати тип метавон, дилхоҳ типи стандарти (дар ҳолати лозими ғайристандарти)-ро истифода кард. Ном ва андозаҳои ин массивҳо низ монанди дигар массивҳо мебошад. Ҳоло тарзи истифодаи ин навъ массивҳоро дар мисоли кучаке дида мебароем.

Масъала. Дар як файли матнӣ k - то массиви дученакаи андозааш ба $M \times N$ баробар дода шудааст. Талаб карда мешавад, барномае нависед, ки матритсаи L -умро ($1 \leq L \leq k$) дар файли дигар сабт кунад.

Ҳал. Ин масъаларо метавон бо роҳҳои гуногун ҳал кард. Яке аз роҳҳои содда, истифодаи массиви сеченака мебошад. Дар ин навъ массивҳо андозаи якум рақами матритса ва ду ченаки боқимонда бошанд, элементҳои матритсаро ифода мекунам. Ба барномаи зерин диққат диҳед:

```
#include <iostream>
#include <fstream>
using namespace std;
int main(){
    ofstream("input.txt","r",stdin);
    freopen("output.txt","w",stdout);
    const int n=3, m=4, k=10;
    int a[k][n][m];
    int l,i,j,f;
    l=2;
    for (f=1; f<=k; f++)
        for (i=0; i<n; i++)
            for (j=0; j<m; j++)
                cin>>a[f][i][j];
    for (i=0; i<n; i++)
    {
        for (j=0; j<m; j++)
            cout<<a[l][i][j]<<" ";
        cout<<endl;
    }
    return 0;
}
```

Функсияи fclose()

Функсияи fclose() барои пӯшонидан ё маҳкам намудани як файл ё stream, ки тавассути функсияи fopen() кушода шудааст, истифода бурда мешавад. Тавре ки қаблан ёдрас карда будем, пеш аз анҷом ёфтани барнома бояд файл баста шавад. Ҳангоми баста нашудани файл сабаби аз даст додани додаҳо ва баъзан ҳатто сабаби нест шудани файл мегардад. Нахви умумии ин функсия намуди зеринро дорад:

int fclose(FILE *fp)

Дар ин ҷо fp низ ҳамон ишорагаре бо файл аст. Дар сурати бомуваффақият баста шудани файл функсия қимати 0 ва дар акси ҳол қимати

гайринулӣ бармегардонад. Ҳангоми мавҷуд набудани файл ё пур будани диск хатогӣ ба вуқӯ меояд.

Функсияи `ferror()`

Функсияи `ferror()` барои ошкор намудани хатогӣҳо дар файли сохташуда истифода бурда мешавад. Наҳви умумии он намуди зеринро дорад:

```
int ferror(FILE *fp)
```

Дар ин ҷо `fp` як ишорагаре ба файл буда, ба қимати `true` доро аст, агар дар амали охирони ба файл гузаронидашуда хатогӣ вучуд дошта бошад ва дар акси ҳол, ба қимати `false` доро мегардад. Барои хатогӣҳои дар файл содиршударо дарёфт кардан бояд пеш аз ҳар амале, ки бо файл карда мешавад, ин функсия даъват карда шавад, вагарна дар охир наметавонад ҳамаи хатогӣҳоро муайян кунад.

Функсияҳои `fscanf()` ва `fprintf()`

Функсияҳои `fscanf()` ва `fprintf()` низ ду функсияи дигари системаи буфердори I/O буда, амали ба функсияи `scanf()` ва `printf()` монандро иҷро мекунанд. Тарзи умумии истифодаи ин функсияҳо ба сурати зерин аст.

```
fprintf(fp, "Control string", argument list)
```

```
fscanf(fp, "Control string", argument list)
```

Дар ин ҷо `fp` ишорагаре аз навъи файл аст, ки тавассути функсияи `fopen()` баргардонида шудааст. Амали ичромекардаи функсияҳо аз он иборат аст, ки қиматҳои худро ба файле, ки тавассути `fp` таъриф шудааст, равона мекунанд.

Мисол. Барномае нависед, ки додаҳои як ширкати мобилӣ, ки дар файл захира шудаанд, пардозिश намояд. Яъне барнома имконият диҳад, ки дар файл рақами муштариӣ нав илова карда шуда, амали ҷустуҷӯ низ иҷро гардад.

```
#include <iostream>
#include <fstream>
#include <stdio.h>
#include <string.h>
#include <ctype.h>
using namespace std;
void add(void);
void lookup(void) ;
char menu(void);

int main(){
    char choice;
    do
    {
        choice=menu();
        switch(choice)
        {
            case 'a' : add(); break;
            case 'l': lookup(); break;
        }
    }
}
```

```

        while(choice!='q');
    }

char menu(void)
{
    char chr ;
    do
    {
        printf("a (Add),l (Lokup, q (Quit) \n");
        chr=tolower(getch());
        printf("\n");
    } while (chr!='q' && chr!='a' && chr!='l');
    return chr;
}

void add(void){
    FILE *fp;
    char name[50];
    int code,num;
    if((fp=fopen("phone","a"))==NULL)
    {
        printf("Cannot open directory File");
        exit(1);
    }
    else
    {
        printf("Enter name and number\n");
        fscanf(stdin,"%s %d",name, &num);
        fscanf(stdin,"%*c");
        fprintf(fp," %s %d \n",name, num);
        fclose(fp);
    }
}

void lookup(void){
    FILE *cpf;
    char name[50], name2[50];
    int code,num,exchg;
    if((cpf=fopen("phone","r"))==NULL)
    {
        printf("Cannot open directory File");
        exit(1);
    }
    printf("name?\n");
    gets(name);
    while(!feof(cpf)){
        fscanf(cpf,"%s %d",name, &num);
        if (!strcmp(name, name2))
        {
            printf("%s %d",name, num);

```

```

        break;
    }
}
fclose(cpf);
}

```

Функсияҳои remove() ва rename()

Функсияи remove() барои аз диск нест кардани як файл истифода бурда мешавад. Нахви истифодабарии он намуди зеринро дорад:

```
int remove( const char *Filename)
```

Дар ин ҷо const char *Filename ҳамон тағйирёбандаи ишорагар аз навъи сохтори FILE мебошад.

Функсияи rename() бошад, барои номи файли дар диск мавҷуд бударо иваз кардан истифода бурда мешавад. Нахви истифодабарии он намуди зеринро дорад:

```
int remove( const char *oldFilename, const char *newFilename)
```

Дар ин ҷо oldFilename, номи мавҷудбуда ва newFilename бошад, номи нави файл мебошад. Ҳангоми бомуваффақият иваз намудани номи файл, функсияи мазкур дорои қимати 0 мегардад, дар акси ҳол бошад, қимати функсия ғайринулӣ аст.

Функсияи fwrite()-ро аслан барои навиштани сохторҳо ва ё мас-сивҳо ба файл истифода мебаранд ва нахви он ба сурати зерин аст:

```
fwrite(array_name,sizeof(array_type),1,fp).
```

Функсияи fread()-ро бошад, барои хондани додаҳои як сохтор ё массив аз файл ба кор мебаранд. Нахви умумии истифодаи он намуди зеринро дорад:

```
fread(array_name,sizeof(array_type),1,fp).
```

Мисол. Дар як файли бинорӣ элементҳои массив захира шудаанд. Барномае нависед, ки ҳамаи элементҳои массивро аз файл хонда, ба саҳифаи тасвир барорад.

```

#include <iostream>
#include <fstream>
using namespace std;
int main(){
    FILE *f;
    int i, n,*a;
    f=fopen("test.dat","rb");
    fread(&n, sizeof(int),1,f);
    cout<<n<<endl;
    a=new int [n];
    fread(a, sizeof(int),n,f);
    for(i=0;i<n;i++)
        cout<<a[i]<<"\t";
    fclose(f);
    return 0;
}

```

Тавре ки қайд карда будем, функцияҳои мазкур ҳам барои хондан ва навиштани массивҳо ва ҳам сохторҳо истифода бурда мешаванд. Ҳоло бошад, ба ин вобаста мисоли дигарро дида мебароем.

Мисол. Барномае нависед, ки элементҳои сохторро аз файл хонда, ба саҳифаи тасвир барорад.

```
#include <iostream>
#include <fstream>
using namespace std;
struct{
    int no;
    char name[25];
    float balance;
} rec;

int main(){
    FILE *fp;
    if((fp=fopen("piple ","rb"))==NULL)
    {
        cout<<"Cannot open File\n";
        exit(1) ;
    }
    while(fread(&rec, sizeof(rec),1,fp)==1)
        cout<<rec.no<<" " <<rec.name<<" " <<rec.balance ;
    fclose(fp);
    return 0;
}
```

Дар ин ҷо оператори `fread(&rec, sizeof(rec),1,fp)` додашуда, ки аз файл хонда мешаванд, дар сохтори `rec` ҷой мегирад, яъне функцияи `fread()` миқдори додашуда, ки аз файл хондааст, бармегардонад, миқдори сохтори додашуда дар аргументи сеюми функцияи мазкур 1 мушаххас карда шудааст. Вақте ки хондан ба охири барнома расид, ин миқдор баробари нол мешавад.

Дар системаи I/O додаҳои сохтор ба намуди бинорӣ захира карда мешаванд. Ва бояд ба таври бинорӣ хонда шаванд. Агар фарз кардему додаҳо дар файл ба намуди матнӣ захира карда шаванд ва дар байни додаҳо адади (дилхоҳ рақам дар ин ҷо барои мисол адади 26 гирифта шуд) бутуни 26 дар қолаби даҳӣ мавҷуд бошад. Аз сабаби он ки файлҳои бинорӣ додаҳо ба намуди байтҳо захира мекунанд. Бинобар ин, агар адади 26-ро ба байтҳо тақсим кунем, намуди зеринро мегирад. $26_{10}=(000000000011010)_2$ вақте ки иҷроиши барнома ба ин ҷо мерасад, рақамҳои сифр ҳамчун EOF фаҳмида шуда, барнома кори худро ба анҷом мерасонад. Ин амал бошад, хатогии бузург ба ҳисоб меравад.

Файлҳои бинорӣ Тавре ки қаблан баён кардем, ин пайдарпайии сохтори додаҳо мебошанд ва додаҳо аз онҳо пайдарпай хонда ва навишта мешаванд. Масалан, дар файли бинорие, ки пайдарпайии ададҳои ҳақиқӣ дар он захира шудаанд, барои хондани 10 адад аз ин пайдарпайи ва баъд як адад ба мо лозим меояд, ки ду маротиба файлро барои хондан кушоем. Ин амал дилгир-кунанда мебошад, зеро барои ҳар амал зарурати аз сари нав кушодан ва бастани файл

пеш меояд. Барои пурратар шинос шудан ба ин амал барномаи зеринро дида мебароем. **Мисол:**

```
#include <iostream>
#include <fstream>
using namespace std;
int main(){
    FILE *f;
    int i, n;
    int a[100];
    f=fopen("test.dat","rb");
    for (i=0;i<10;i++)
        fread(a, sizeof(int),10,f);
    fclose(f);
    for(i=0;i<10;i++)
        cout<<a[i]<<"\t";
    f=fopen("test.dat","rb");
    fread(&n, sizeof(int),1,f);
    cout<<n<<"\t";
    fclose(f);
    return 0;
}
```

Чи хеле ки дида мешавад, барои хондани 10 адад аз файл як маротиба файл кушода шуда, пас аз хондани ададҳо баста шудааст. Барои аз сари нав хондани элементи якум, бори дигар ичрои амали кушодан ва пӯшидани файл лозим шудааст. Ин амал ҳарчанд, ки ичро мешавад, вале агар онро тавассути функсияи `fseek()` анҷом диҳем, хеле сода ичро мегардад.

Функсияи `fseek()`

Функсияи мазкур мавқеи файл ё ишорагари мавқеи файло метавонад муайян созад. Бо ёрии ин функсия ишорагаре ба мавқеи лозимии файл равон карда мешавад. Наҳви умумии истифодаи функсияи мазкур намуди зеринро дорад:

```
int fseek(FILE *fp, long int num_инт, int origin);
```

Дар ин ҷо `fp` ишорагаре аз навъи сохтори `FILE` аст, ки бо фурӯхонии функсияи `fopen()` баргардонидида шудааст, `num_инт`-миқдори ҷустуҷӯшаванда аз мабдаъ ё `origin` мебошад, ки `origin` дар навбати худ яке аз макросҳои ҷадвали зерин мебошад.

	Номи макрос	Мабдаъ ё <code>origin</code>
	<code>SEEK_SET</code>	Шуруъ аз аввали файл
	<code>SEEK_CUP</code>	Шуруъ аз мавқеи ҷорӣ
	<code>SEEK_END</code>	Шуруъ аз интиҳои файл

Ин макросҳо дорои миқдори бутун низ мебошанд. Масалан, дар мавриди `SEEK_SET` 0 дар мавриди `SEEK_CUP` 1 ва дар мавриди `SEEK_END` 2 .

Агар қимати `num_инт` мусбат бошад, ишорагари файл додаҳоро аз мавқеи нишондодашуда ба пеш мехонад ва дар сурати манфӣ будан додаҳоро аз мавқеи нишондодашуда ба қафо мехонад.

Истифодаи функсияи `fseek()` барои файлҳои матнӣ тавсия дода намешавад, зеро он махсус барои файлҳои бинорӣ пешбинӣ шудааст ва дар сурати файлҳои матнӣ барнома ба хатоӣ рӯбарӯ мешавад.

Мисол. Дар як файли бинорӣ элементҳои массиви навъи бутун дода шудааст. Барномае нависед, ки ҷойи элементҳои максималӣ ва минималии массивро дар дохили файл иваз кунад.

```
#include <iostream>
#include <fstream>
using namespace std;
int main(){
    FILE *f;
    int i, n, nmax, nmin, max, min;
    double *a;
    f=fopen("simple.dat","rb+");
    fread(&n, sizeof(int),1,f);
    cout<<"n="<<n<<endl;
    a=new double[n];
    fread(a, sizeof(double),n,f);
    for (i=0;i<n;i++)
        cout<<a[i]<<"\t";
    nmax=nmin=0 ;
    max=min=a[0];
    for(i=1;i<n;i++)
    {
        if (a[i]>max)
        {
            max=a[i];
            nmax=i;
        }
        if (a[i]<min)
        {
            min=a[i];
            nmin=i;
        }
    }
    fseek(f, sizeof(int)+nmax*sizeof(double),SEEK_SET);
    fwrite(&min, sizeof(double),1,f);
    fseek(f, sizeof(int)+nmin*sizeof(double),SEEK_SET);
    fwrite(&max, sizeof(double),1,f);
    fclose(f);
    delete [] a;
    return 0;
}
```

Саволҳо барои мустаҳкамкунӣ

1. Барои ҷӣ дар барноманависӣ файлҳои лозиманд?
2. Ҷараён чист ва чанд хел мешавад?
3. Функсияҳои буфердор ҷӣ гуна функсияанд?
4. Дар забони C/C++ чанд намуди файлҳои ба кор бурда мешаванд?

5. Чараёнҳои ifstream ва ofstream чӣ гуна чараёнанд?
6. Байни файлҳои матнӣ ва бинорӣ чӣ хосияти фарқкунанда мавҷуда аст?
7. Функсияҳои fprintf() fscanf() чӣ гуна функсияанд?
8. Барои охири як файло муайян кардан кадом функсия истифода бурда мешавад?
9. Бастанӣ файл барои чӣ лозим аст ва кадом функсия барои бастанӣ як файл истифода бурда мешавад?
10. Барои номи як файло иваз кардан аз кадом функсия истифода бурда мешавад?
11. Магар як маълумотро яку якбора дар якҷанд файл навиштан мумкин аст?
12. Оё метавон қисме аз элементҳои массивро дар файл ва қисмеро дар саҳифаи тасвир навишт?
13. Элементҳои массивро оё метавон аз якҷанд файл хонд?
14. Функсияи fseek() барои чӣ истифода бурда мешавад ва он дорои чанд тасро мебошад?
15. Функсияҳои fread() ва fwrite() барои чӣ хизмат мекунанд?
16. Функсияи feof() барои чӣ лозим аст ва он кадом вазифаро иҷро мекунад?
17. Кадом функсия барои муайян намудан ё ошкор намудани хатогиҳо дар файл истифода бурда мешаванд?
18. Оё тамоми элементҳои як сохторро метавон яку якбора ба як файл навишт?
19. Ҳангоми бомуваффақият кушода нашудани файл чӣ ҳодиса руи медиҳад?
20. Ба фикри Шумо кадом навъи файлҳо барои захира кардани додаҳо мувофиқтар аст?

Намунаи ҳалли масъалаҳо

1) Ҳама элементҳои матритсаи квадратии андозааш ба $N \times N$ баробар аз сифрҳо (0) ва воҳидҳо (1) таркиб ёфтаанд. Миқдори «чазираҳо»-е, ки аз воҳидҳо иборатанд ёфта шаванд. Тахти мафҳуми «чазира» гурӯҳи воҳидҳо (ё як воҳид) фаҳмида мешавад, ки аз ҳамаи тарафҳо бо сифрҳо ихота карда шудааст. (ё бо тарафҳои матриса). Гурӯҳи воҳидҳо як чазираро ташкил медиҳанд, агар аз яктои онҳо ба катакҷаи ҳамсоя бе убури катакҷае, ки дар он сифр аст, гузаштан мумкин бошад. Катакҷаҳо ҳамсоя номида мешаванд, агар дар самти амуди ё уфуқии якдигар ҷойгир бошанд.

Додаҳои вурудӣ

Дар сатри нахустини файли input.txt адади натуралии N (андозаи матриса), ки аз 100 зиёд нест навишта мешавад. Дар N сатри боқимонда элементҳои матритса тавассути фосила (пробел) навишта мешаванд.

Додаҳои хуруҷӣ

Дар файли output.txt як адад- миқдори чазираҳо хорич карда мешавад.

Мисол

input.txt	output.txt
-----------	------------

5	4
1 0 1 1 1	
0 0 0 0 0	
1 1 1 0 1	
0 1 0 0 1	
0 0 0 1 1	

Матни барнома.

```
#include <bits/stdc++.h>
using namespace std;
char a[105][105];
bool used[105][105];
void dfs(int x, int y){
    used[x][y] = true;
    if(a[x+1][y] == '1' && !used[x+1][y]) dfs(x+1, y);
    if(a[x-1][y] == '1' && !used[x-1][y]) dfs(x-1, y);
    if(a[x][y+1] == '1' && !used[x][y+1]) dfs(x, y+1);
    if(a[x][y-1] == '1' && !used[x][y-1]) dfs(x, y-1);
}
int main(){
    int n;
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    cin >> n;
    for(int i=1; i<=n; ++i){
        for(int j=1; j<=n; ++j){
            cin >> a[i][j];
        }
    }
    int ans = 0;
    for(int i=1; i<=n; ++i){
        for(int j=1; j<=n; ++j){
            if(a[i][j] == '1' && !used[i][j]){
                dfs(i, j), ++ ans;
            }
        }
    }
    cout << ans << endl;
    return 0;
}
```

2) Рассоми машхур хост то шоҳқории нав созад. Пас аз чанд рӯзи кори хост то заҳмати (санъати) худро таҳқиқ кунад Рассом ба ёд овард, ки ба сурати зерин расм тасвир карда шудааст: ибтидо матоъи сафед (доска), ки шакли росткунчавии паҳноияш ба w ва баландиаш ба h баробар аст мегирад. Сипас, дар ин матоъи сафед n росткунча, ки тарафҳояш ба тарафи матоъи сафед параллел буда, қуллаҳояш дорои координатаҳои бутун мебошанд, тасвир

мекунад. Ба рассом кумак кунед, то масоҳати ранг накардаи матоъи сафедро муайян кунед.

Додаҳои вурудӣ

Дар сатри нахустини файли вурудии INPUT.TXT ду адади натуралии w ва h ($1 \leq w, h \leq 100$) ва дар сатри дуюм адади бутуни n ($0 \leq n \leq 5000$) – миқлори росткунҷаҳо навишта мешавад. Дар n -сатри боқимонда бошад, маълумот оиди ҳамаи росткунҷаҳо навишта мешавад. Ҳар як сатр маълумоти як росткунҷаро ба шакли чор адад x_1, y_1, x_2, y_2 ифода мекунад. Дар инҷо (x_1, y_1) ва (x_2, y_2) – мувофиқан координатаҳои кунҷҳои рости болоӣ ва чапи поёнии росткунҷаро ифода мекунанд.

Додаҳои хуруҷӣ

Дар файли хуруҷии output.txt як адади бутун – масоҳати қисми рангнокардаи муқаллам хорич карда мешавад.

Мисол

№	input.txt	output.txt
1	5 5 2 1 1 3 3 2 2 4 4	18
2	6 7 3 0 0 5 5 1 1 4 4 2 2 3 3	17

Матни барнома.

```
#include <bits/stdc++.h>
using namespace std;
bool used[105][105];
int main(){
    int n, m, k;
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    cin >> n >> m >> k;
    swap(n, m);
    for(int a=1; a<=k; ++a){
        int x1, y1, x2, y2;
        cin >> x1 >> y1 >> x2 >> y2;
        for(int i=x1; i<x2; ++i){
            for(int j=y1; j<y2; ++j){
                used[i][j] = true;
            }
        }
    }
    int ans = 0;
    for(int i=0; i<n; ++i){
```

```

        for(int j=0; j<m; ++j){
            ans += (1 - used[i][j]);
        }
    }
    cout << ans << endl;
    return 0;
}

```

3) Миқдори ҳамаи ададҳои N-рақама дар системаи ҳисоби K, ки дар навишташон ду сифр(0) пайдарпай вонамехурад, ҳисоб карда шавад.

Додаҳои вурӯдӣ

Дар файли input.txt ду адади натуралии N ва K дар системаи ҳисоби даҳӣ навишта мешаванд. ($2 \leq K \leq 10$; $2 \leq N$; $4 \leq N+K \leq 18$).

Додаҳои хуруҷӣ

Дар файли output.txt як адади бутун дар системаи ҳисоби даҳӣ-ҷавоби мисол хориҷ карда шавад.

Мисолҳо

№	input.txt	output.txt
1	2 10	90
2	4 2	5
3	6 3	328

Матни барнома.

```

#include <bits/stdc++.h>
using namespace std;
char a[18], b[25];
int ans = 0;
void rec(int n, int m, int k){
    if(n == 0){
        ++ ans;
        return;
    }
    int i = 0;
    if(n == m || b[n+1] == '0')
        i = 1;
    for(; i<k; ++i){
        b[n] = a[i];
        rec(n-1, m, k);
    }
}
int main(){
    int n, k;
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    cin >> n >> k;
    a[n+1] = '1';
    for(int i=0; i<10; ++i)

```

```

a[i] = i + '0';
for(int i=10; i<18; ++i)
a[i] = (i - 10) + 'A';
rec(n, n, k);
cout << ans << endl;
return 0;
}

```

4) Шоҳи Флатландия хост то якчанд дарахти назди қасрашро бурад. Дарахтон дар назди қасри шоҳ дар як қатор шинонида шудаанд, ки миқдори умумии онҳо ба n баробар аст. Масофаи байни дарахтон якхела мебошад.

Баъди буридан дар назди қаср бояд m дарахт боқи монда, ки масофаи байни онҳо низ якхела бошад. Ба шоҳ кумак кунед, яъне миқдори имконпазири усулҳои буридани дарахтонро ба \bar{y} фаҳмонед.

Барномае нависед, ки бо дода шудани ададҳои n ва m миқдори усулҳои буридани якчанд дарахтро аз байни n дарахт муайян кунад, ки m то боқӣ монад ва масофаи байни онҳо якхела бошад.

Додаҳои вурудӣ

Дар файли input.txt ду адади бутуни n ва m ($0 \leq m, n \leq 1000$) навишта меҳӯшавад.

Додаҳои хуруҷӣ

Дар як сатри файли output.txt як адади бутун—миқдори усулҳо.

Мисол

№	input.txt	output.txt
1	5 3	4

Қайд. Агар шартан ададҳои дар назди қаср руидаро бо ҳарфҳо ба шакли «TTTT» ишорат кунем, онгоҳ нас аз буридан дарарахтони боқимонда метавонанд чунин шакл дошта бошанд: «TTT..», «.TTT.», «..TTT», «T.T.T».

```

#include <bits/stdc++.h>
using namespace std;
int main(){
int n, m;
freopen("input.txt", "r", stdin);
freopen("output.txt", "w", stdout);
cin >> n >> m;
int ans = 0;
for(int i=0; i<n; ++i){
int k = (m - 1) * i + m;
if(k <= n) ans += (n - k + 1);
}
cout << ans << endl;
return 0;
}

```

5) **Масъала.** Барномае нависед, ки бо додаҳои як файл, ки аз маъ-лумотҳои коргарон иборатанд, чунин амалиётро иҷро кунад:

1. Хондан аз файл
2. Сабт кардан дар файл
3. Илова намудани коргари нав
4. Тағйир додани додаҳои коргар
5. Нест кардани коргар аз файл
6. Дар саҳифаи тасвир намоиш додани додаҳои коргар
7. Баровардани ҳисоботи коргарон

Барои ҳалли ин масъала сохторе, ки майдонҳои он аз код, ном, насаб, маъош ва синну сол иборат мебошанд, ба намуди зерин муаррифӣ карда мешавад:

```
struct StructKorgar {
    int kod;
    char name[15];
    char LastName[10];
    long int maosh;
    int sol;
}
```

Баъд аз ин массиве бо номи korgaron аз навъи сохтор муаррифӣ карда мешавад.

Ба ғайр аз ин, ҳафт функсия, ки пешнамунаи онҳо намуди зеринро дорад, муаррифӣ карда мешаванд:

```
void namoish_dodan (int n);
void khondan_az_file (int n);
void sabt_kardan (int n);
int ilova_kardan (int n);
void tahrir_kardan (int n);
int nest_kardan (int n);
void hisobot (n);
```

Биёед, аввал ҳар яке аз ин функсияҳоро алоҳида-алоҳида дида мебароем.

Функсияи namoish_dodan (int n), функсияи мазкур барои дар саҳифаи тасвир намоиш додани додаҳои аз файл хондашуда хизмат мекунад.

Функсияи khondan_az_file (int n) барои аз файл хондани додаҳои коргарон ва тавассути функсияи namoish_dodan (int n) дар саҳифаи тасвир навиштани додаҳо истифода бурда мешавад.

Функсияи sabt_kardan (int n) барои сабт намудани додаҳо дар файл хизмат мекунад.

Функсияи ilova_kardan(int n) барои илова намудани коргари нав истифода бурда мешавад.

Функсияи tahrir_kardan (int n) барои таҳрир намудани додаҳои n-ум коргар истифода бурда мешавад.

Функсияи nest_kardan (int n) барои нест кардани додаҳои коргар n-ум истифода бурда мешавад.

Функсияи hisobot (n) бошад, барои ҳамаи додаҳои коргаронро дар саҳифаи тасвир намоиш додан истифода бурда мешавад.

Акнун, ки ба функсияҳои лозимӣ шинос шудем, биёед матни барномаро ба таври пурра меорем.

Матни барнома.

```
#include <iostream>
#include <fstream>
#include <iomanip>
using namespace std;
int n;
struct StructKorgar{
    int kod;
    char name[15];
    char LastName[10];
    long int maosh;
    int sol;
} korgaron[100];

void namoish_dodan (int n){
    int i;
    if (n != 0)
    {
        for (i = 0; i < n; i++)
        {
            cout << "\t-----";
            cout << "\n\tKod: " << korgaron[i].kod;
            cout << "\n\tname: " << korgaron[i].name;
            cout << "\n\tLastName: " << korgaron[i].LastName;
            cout << "\n\tMaosh: " << korgaron[i].maosh;
            cout << "\n\tSol: " << korgaron[i].sol << "\n";
        }
        cout << "\t-----";
    }
}

void khondan_az_file (int n){
    int i;
    ifstream fin ("korgaron.txt");
    fin >> n;
    for (i = 0; i < n; i++)
    {
        fin >> korgaron[i].kod;
        fin >> korgaron[i].name;
        fin >> korgaron[i].LastName;
        fin >> korgaron[i].maosh;
        fin >> korgaron[i].sol;
    }
    namoish_dodan (n);
    cout << "\n\n\tbarovardan...";
}

void sabt_kardan (int n){
    int i;
    ofstream fout ("korgaron.txt");
```

```

    fout << n << "\n";
    for (i = 0; i < n; i++){
        fout << korgaron[i].kod << "\t";
        fout << korgaron[i].name << "\t";
        fout << korgaron[i].LastName << "\t";
        fout << korgaron[i].maosh << "\t";
        fout << korgaron[i].sol << "\n";
    }
    cout << "\tSabt...";
}

int ilova_kardan (int n){
    cout << "\tKod: " << n + 1;    korgaron[n].kod = n + 1;
    cout << "\n\tName: ";    cin >> korgaron[n].name;
    cout << "\tLastName: ";    cin >> korgaron[n].LastName;
    cout << "\tMaosh: ";    cin >> korgaron[n].maosh;
    cout << "\tSod: ";    cin >> korgaron[n].sol;
    n++;
    cout << "\n\tIlova ...";
    return n;
}

void tahrir_kardan (int n){
    int tmp;
    namoish_dodan (n);
    cout << "\n\n";
    cout << "\tKodi korgarro baroi tahrir kardan dohil kuned : ";
    cin >> tmp;
    if (tmp != 0)    {
        tmp--;
        cout << "\n\tKod: " << tmp + 1;    korgaron[tmp].kod = tmp + 1;
        cout << "\n\tName: ";    cin >> korgaron[tmp].name;
        cout << "\tLastName: ";    cin >> korgaron[tmp].LastName;
        cout << "\tMaosh: ";    cin >> korgaron[tmp].maosh;
        cout << "\tSol: ";    cin >> korgaron[tmp].sol;
        cout << "\n\tTahrir\n\n";
        namoish_dodan (n);
    }
}

int nest_kardan (int n){
    int i, tmp;
    namoish_dodan (n);
    cout << "\n\n";
    cout << "\tKodi korgarro baroi nest kardan dohil kuned: ";
    cin >> tmp;
    if (tmp != 0)    {
        tmp--;
        for (i = tmp; i < n; i++)
            {

```

```

        korgaron[i] = korgaron[i+1];
        korgaron[i].kod = i + 1;
    }
    n--;
    cout << "\n\tNest kardan\n\n";
    namoish_dodan (n);
    return n;
}
}

void hisobot (int n) {
    int i, phoiz;
    float daromad;
    cout << " -----\n";
    cout << " | LastName | Sol | Maosh | % | Daromad |\n";
    cout << " -----\n\n";
    for (i = 0; i < n; i++) {
        phoiz = 5;
        if (korgaron[i].maosh >= 1000){
            phoiz++;
        }
        if (korgaron[i].sol <= 1999){
            phoiz++;
        }
        daromad = korgaron[i].maosh / phoiz;
        cout << " -----\n";
        cout << " |" << setw(16) << korgaron[i].LastName
            << " |" << setw(6) << korgaron[i].sol
        << " |" << setw(8) << korgaron[i].maosh << setw(5) << "somoni"
            << " |" << setw(5) << phoiz << "%"
            << " |" << setw(6) << daromad << setw(5) << "somoni"
            << " |\n";
    }
    cout << "-----\n";
}

int main(){
    int n, tmp;
    ifstream fin ("korgaron.txt");
    fin >> n;
    menu :
    cout << "1. Hondan az fayl" << endl;
    cout << "2. Dar fayl sabt kardan" << endl;
    cout << "3. Ilova namudani korgar" << endl;
    cout << "4. Tagiir dodan" << endl;
    cout << "5. Nest kardani korgar" << endl;
    cout << "6. Namosh dodani korgar" << endl;
    cout << "7. Barovardani hisobot" << endl;
    cout << "8. Baromad" << endl;
    cout << "\n\n" << "Amaliyotro dohil kuned: ";
}

```

```

cin >> tmp;
cout << "\n";
switch (tmp)
{
case 1: khondan_az_file (n); cout << "\n\n";
        goto menu;
case 2: sabt_kardan (n); cout << "\n\n";
        goto menu;
case 3: if (korgaron[0].kod == 0 || n == 0) {
            n = ilova_kardan (0); cout << "\n\n";
            goto menu;
        }
        else
        {
            n = ilova_kardan (n); cout << "\n\n";
            goto menu;
        }
};
case 4: if (korgaron[0].kod == 0 || n == 0) {
        cout << "\tBD holi...\n\n";
        goto menu;
    }
    else {
        tahrir_kardan (n); cout << "\n\n";
        goto menu;
    }
};
case 5: if (korgaron[0].kod == 0 || n == 0)
{
        cout << "\tBD holi...\n\n";
        goto menu;
    }
    else
    {
        n = nest_kardan (n);
        cout << "\n\n";
        goto menu;
    }
};
case 6: if (korgaron[0].kod == 0 || n == 0)
{
        cout << "\tBD holi...\n\n";
        goto menu;
    }
    else {
        namoish_dodan (n); cout << "\n\n";
        goto menu;
    }
};
case 7: if (korgaron[0].kod == 0 || n == 0){
        cout << "\tBD holi...\n\n";
        goto menu;
    }
    else

```

```

        {
            hisobot (n); cout << "\n\n";
            goto menu;
        };
    case 8: break;
    default: break;
}
return 0;
}

```

Масъалаҳо барои кори мустақилона

1. Файли матнӣ дода шудааст, пеш аз ҳар сатри он рақами сатрро гузored.
2. Ду адади натуралӣ дар файл дода шудаанд. Қимати миёнаи арифметики ва геометрии онҳоро ёбед.
3. Радиуси давра R дар файл дода шудааст. Дарозии давра ва масоҳати доираи бо ин давра маҳдудро ҳисоб кунед.

$$L = 2\pi R, S = \pi R^2$$

4. Адади серақама дар файл додашудааст. Ҷойи разрядҳои садӣ ва даҳии онро иваз кунед.
5. Адади серақама дар файл дода шудааст. Қимати миёнаи арифметики ва геометрии рақамҳои ададро ҳисоб кунед.
6. Дарозии тарафҳои секунҷа a, b, c дар файл дода шудааст. Барномае нависед, ки бо истифода аз формулаи Герон масоҳати онро ҳисоб кунед.

Қайд. *Формулаи Герон намуди зеринро дорад,*

$$S = \sqrt{r(r-a)(r-b)(r-c)}$$

ки дар ин ҷо r -нимпериметр мебошад. Ва аз геометрия маълум аст, ки $P=a+b+c$ $var = \frac{P}{2} = \frac{a+b+c}{2}$

7. Файле иборат аз ададҳои бутун дода шудааст. Дар ҳар сатри файл адади максималиро ёфта, дар файли нав нависед.
8. Файл дода шудааст, ки дар он матне иборат аз якчанд сатр навишта шудааст. Барномае нависед, ки дар охири ҳар сатр миқдори ҳарфҳои ҳар сатрро нависад.
9. Файл иборат аз ададҳои бутун дода шудааст. Дар ҳар сатри файл миқдори ададҳои манфиро ҳисоб намуда дар файли нав нависед.
10. Ду файли матнӣ дода шудааст. Дар файли сеюм ҳамаи ададҳоеро, ки дар файли якум мавҷуданду дар файли дуввум нестанд, нависед.
11. Файл дода шудааст, ки аз таърихи рӯзҳо иборат мебошад. Масалан, 25.04.1989 . Таърихи рӯзи минималиро ёбед.
12. Файли матнӣ дода шудааст, ки дар он ададҳои бутун навишта шудаанд. Барномае нависед, ки ...-и онро ҳисоб кунед.
 - ✓ миқдори ададҳои чуфт ва тоқ
 - ✓ миқдори ададҳои мусбат ва манфӣ
 - ✓ миқдори ададҳои сода ва мураккаб
 - ✓ миқдор ва суммаи ададҳои мукамал
 - ✓ қимати миёнаи арифметикӣ ва геометрии ададҳо

- ✓ адади максималӣ ва минималӣ
13. Дар як файли матнӣ калимаҳои русӣ бо тарҷумаи тоҷикӣ навишта шудааст. Барномае нависед, ки бо дохил кардани калима тарҷумаи онро барорад.
 14. Ададҳои натуралии n ва m дар файли матнӣ дода шудааст. Барномаи ёфтани калонтарин тақсимкунанда (КТУ) – и онҳоро созед.
 15. Файли матнӣ, ки дар он ададҳои бутун пайиҳам навишта шудаанд, дода шудааст. Элементи максималиро дар ҳар сатр ёбед.
 16. Файли матнӣ дода шудааст. Миқдори сатрҳои холии онро ҳисоб кунед.
 17. Файли матнӣ дода шудааст. Дар файли дигар калимаи кутохтарини онро нависед.
 18. Адади бутуни N ва гурӯҳи ададҳои бутуни миқдорашон ба N баробар, ки дар файли матнӣ ҷойгиранд, дода шудааст. Миқдори ададҳоеро, ки дар байни аввалин ва охирин адади калонтарин ҷойгиранд, ёбед. Агар дар гурӯҳи ададҳо танҳо як адади калонтарин мавҷуд бошад, он гоҳ 0 хориҷ карда шавад. Натиҷаи амалиётҳо дар файли мавҷудбуда сабт карда шавад.
 19. Массиви бутуни андозааш N , ки дар файли матнӣ ҷойгир аст, дода шудааст. Муайян карда шавад, ки оё дар массив ададҳои ҷуфт ва тоқ пайдарпай ҷойгир шудаанд ё не. Агар ададҳои ҷуфт ва тоқ пайдарпай ҷойгир бошанд, он гоҳ 0 ва дар ҳолати баръакс рақами тартибии аввалин адади пайдарпайро вайронкунанда хориҷ карда шавад. Натиҷаи амалиётҳо дар файли мавҷудбуда сабт карда шавад.
 20. Файли матнӣ (сатрӣ) дода шудааст. Файли нав сохта шавад, ки дар он сатрҳои файли додашуда ба таври лексикографӣ, яъне сатрҳо бо тартиби зиёдшавии коди рамзҳоишон (аз рамзи аввалин сар карда) ҷойгир шудаанд.
 21. Ададҳои бутуни x_1, x_2, \dots, x_n ки шумораашон n -то мебошад, дар файли матнӣ дода шудааст. Дар байни онҳо аломатҳои "+" ва "-" –ро тавре гузоред, ки қимати ифодаи сохташуда ба адади бутуни додашудаи S баробар бошад. Агар чунин амал ғайриимкон бошад, он гоҳ сатри "No solution" ва дар ҳолати баръакс ифодаи тартибдодашуда дар файли нав хориҷ карда шавад. Агар чунин ифодаҳои тартибдодашуда, ки қиматашон ба S баробар аст, якчандто бошанд, он гоҳ яке аз онҳо дар файл хориҷ карда шавад.
 22. Адади натуралии n дода шудааст. Ададҳои $1, 2, \dots, n^2$ –ро аз рӯи спирал дар файл нависед.
 23. Дар пайдарпайии ададҳои a_1, a_2, \dots, a_3 аъзои якум дода шудааст. Аъзоҳои боқимондаро аз рӯи формулаи $a_i = (a_{i-1})^2 \bmod 10000$ ҳисоб карда дар файл сабт кунед. Сипас, аъзои n -уми онро хориҷ кунед.
 24. Дар файли матнӣ матне дода шудааст. Барномае нависед, ки миқдори калимаҳо, миқдори сатрҳо, миқдори калимаҳои такроршавандаро дар намуди ҷадвали содиротӣ дар экран хориҷ кунед.
 25. Дар файли матнӣ ду ҷумла дода шудааст. Муайян карда шавад, ки оё калимаҳое, ки ҷумлаи якумро ташкил додаанд дар ҷумлаи дуюм дида мешаванд ё не.
 - 26. Масъалаҳои боби массивҳоро бо истифод аз файлҳо ҳал кунед.**

Боби 11. Барноманависии динамикӣ

Муқаддима

Барноманависии динамикӣ ин усули барноманависие мебошад, ки дар он барномаҳои калон ба қисматҳои хурд тақсим карда шуда, ба намуди соддатар ҳал карда мешаванд. Ин навъ барноманависӣ дар назарияи идоракунӣ, назарияи системаҳои ҳисобкунӣ ва ғайра истифода бурда мешавад.

Сабаби динамикӣ ном гирифтани ин навъ барноманависӣ дар он аст, ки қариб ҳамаи додаҳо (тағйирёбандаҳо) ба намуди динамикӣ муаррифӣ гардида истифода бурда мешаванд.

Дар боби ишорагарҳо баъзе аз истифодаи додаҳои динамикиро дида баромада будем. Дар ин боб, қариб ҳамаи он мафҳумро истифода мекунем.

Воситаҳои муҳимтарини ин навъ барноманависӣ дар поён оварда шудаанд:

- 1) Листҳои пайвандӣ;
- 2) Пушаҳо (стек);
- 3) Сафҳо (очеред);
- 4) Графҳо;
- 5) Дарахтҳои дуй.

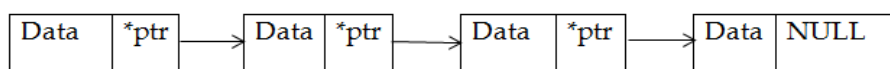
Ҳар яке аз ин воситаҳо мавриди баҳс қарор медиҳем.

Листҳои пайвандӣ

Листҳои пайвандӣ як навъе аз сохтори додаҳои динамикӣ мебошанд, ки онҳо доимӣ набуда, дар рафти иҷроиши барнома метавонанд андозаи худро дар хотираи компютер тағйир диҳанд. Ҳар як элементи листи пайвандӣ гирех (node) номида шуда, дорои ду элемент мебошад, ки элементи аввала ягон додари ифода намуда, элементи дуввум ба гирехи баъди ишора мекунад. Ба сифати элементи аввала метавон дилҳо додари (тағйирёбандаҳои оддӣ, массивҳо, сохторҳо, объектҳо ва ғайра) истифода кард.

Ҳар як элементи листи пайвандӣ суроғи элементи листи дигарро дар худ нигоҳ медорад. Ба ин минвол метавон ба ҳамаи гирехҳои листи пайвандӣ дастрас шуд.

Баъзан листи пайвандиро инчунин листҳои динамикӣ низ мегӯянд. Сабаби пайвандӣ ном бурдани онҳо дар он аст, ки ҳар як гирех бо гирехҳои дигар пайванд мебошад. Сабаби динамикӣ гуфтани ин навъ додаҳо дар он аст, ки андозаи онҳо пас аз иҷроиши барнома метавонад ба таври дилҳо тағйир ёбанд. Барои ба хонанда пурратар шинос шудан дар поён листи пайвандии дорои чор гирехро ба намуди графикӣ меорем. Расми 1.

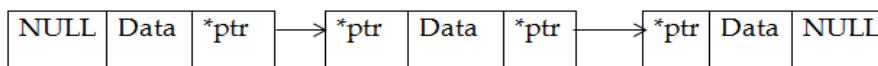


Расми 1.

Тавре ки аз ин расм дида мешавад, ҳар як гирехи листи пайвандӣ дорои ду элемент иттилоот ва суроға мебошад. Ҳар як гирех суроғи гирехи баъдинаро дар худ нигоҳ медорад. Гирехи листи охирон ба ҷое ишора намекунад ва қимати қисми суроғи он NULL (сифр) мебошад. Листеро, ки ҳар гирехи он бо

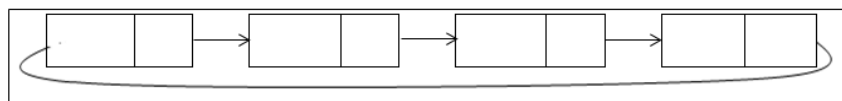
як пайванд ба гиреҳи баъди пайваст мешавад, листи якпайвандӣ номида мешавад.

Баъзе аз листи пайвандӣ ҳастанд, ки ҳар гиреҳи онҳо дорои ду пайванд буда, як пайванди он ба гиреҳи баъди ва пайванди дигарашон ба гиреҳи қабли ишора мекунад. Ин навъ листҳоро листҳои ду пайвандӣ меноманд. Дар ин навъ листҳо ишорагари гиреҳи аввал ва охир ба ҷое ишора намекунанд, ва қимати қисми суроғайи онҳо NULL мебошад. Барои ба хонанда пурратар намудани ин матлаб дар поён листи дупайвандии дорои се гиреҳ оварда шудааст.



Расми 2.

Илова бар ин навъи дигари листи пайванди низ вуҷуд доранд, ки онҳоро листи пайвандии ҳалқавӣ меноманд. Дар ин навъ листи пайванди қисми ишорагари гиреҳи охирон ба гиреҳи аввала ишора мекунад. Листҳои пайвандии ҳалқавӣ метавонанд, листҳои якпайванди ё ду пайвандӣ бошанд. Листҳои пайвандии ҳалқавӣ шакли зеринро доранд:



Расми 3.

Барои ба моҳияти листҳои пайванди хубтар шинос шудан аввал листҳои якпайвандӣ ва баъд листҳои дупайвандиро мавриди баҳс қарор медиҳем.

Эълон намудани Node

Пеш аз кор кардан ба листҳои пайвандӣ зарур аст, ки навъе барои гиреҳҳои он муаррифӣ карда мешавад. Гиреҳҳои листи пайвандӣ асосан ба намуди сохтор муаррифӣ карда мешаванд. Наҳви муаррификунии сохтори Node шакли зеринро дорад:

```
struct номи сохтори Node
{
    Навъ итилоот;
    Номи сохтор *ишорагар;
}
```

Дар ин ҷо ба сифати итилоот, метавон додаи дилхоҳ навъи стандартӣ ва ғайристандартиро истифода кард. Ба сифати мисол сохтори Node-ро дида мебароем, ки дорои ду майдон мебошад. Майдони якуми он адади бутун буда, майдони дуввум ишорагар мебошад.

```
struct Node{
    int data;
    Node *next
}
```

Пас аз ин дастур метавон гиреҳҳои лозимиро аз навъи ин сохтор ба сурати зерин муаррифӣ кард.

```
Node *p,*q;
```

Бо ёрии ин дастур ду гиреҳ муаррифӣ карда мешавад.

Пайваст намудани гиреҳҳои листҳои якпайвандӣ

Барои эҷод намудани листҳои пайвандӣ бояд, гиреҳҳои он ба ҳамдигар пайваст карда шаванд. Ишорагарҳои *p ва *q –ро, ки қаблан муаррифӣ намуда будем, дар назар гирифта, қитъаи барномаи зеринро менависем.

```
p=new Node;  
p->data =1;  
q=new Node;  
q->data =2;
```

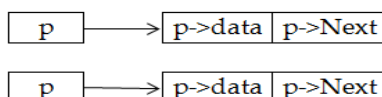
Бо иҷрои ин дастурҳо ду node барои листи пайванди эҷод карда мешавад. Пас аз эҷод намудани node-ҳо зарурати пайваст намудани онҳо пеш меояд. Барои пайваст намудани гиреҳҳои p ва q аз дастурҳои зерин истифода бурда мешавад.

```
p->Next =q;
```

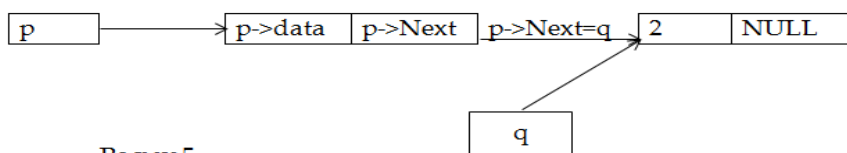
Маънои ин дастур ин аст, ки ҷое, ки q ба онҷо ишора мекунад p->Next низ бояд ба онҷо ишора кунад. Бо суханҳои дигар ишорагари гиреҳи p ба гиреҳи q ишора мекунад. Азбаски ишорагари Next –и гиреҳи q ба ҷое ишора намекунад, ба он қимати NULL мебахшем. Барои иҷрои ин амал яке аз роҳҳои зерин истифода бурда мешавад:

- 1) q->Next =NULL;
- 2) p->Next->Next =NULL;

Маънои дастури якум ин аст, ки ба майдони Next-и гиреҳи q қимати NULL бахшида шавад ва маънои дастури дуввум ин аст, ки ба майдони Next-и node ки ишорагари p ба он ишора мекунад қимати NULL бахшида шавад. Барои моҳияти ин дастурҳоро фаҳмидан онҳоро ба намуди графикӣ тасвир мекунем. Дар натиҷа шакли зерин ҳосил мешавад.



Расми 4.



Расми 5.

Илова намудани node дар листи пайвандӣ

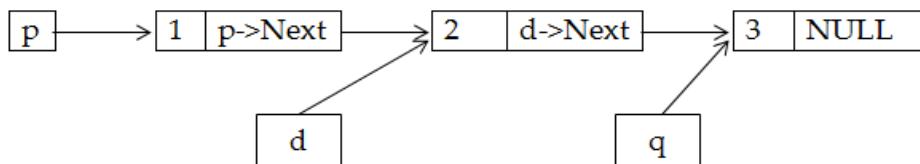
Дар расми 5 листи пайвандии дорои ду гиреҳҳои оварда шудааст. Дар онҷо p-ро ибтидои лист ва q-ро интиҳои лист меноманд. Илова бар ин метавон ба листи пайвандӣ гиреҳҳои нав низ ҳамроҳ кард. гиреҳҳои нав метавонанд дар аввал, дар мобайн ё дар охири листи пайвандӣ қарор гиранд. Масалан, барои дар байни гиреҳҳои p ва q илова намудани гиреҳи нав аввал онро ба сурати зерин эҷод мекунем:

```
Node *d=new Node;  
d->data=3;
```

Пас аз сохта шудани гиреҳи нав зарурати онро дар байни гиреҳҳои p ва q гузоштан пеш меояд. Барои иҷрои ин амал дастурҳои зерин истифода бурда мешавад:

```
p->Next=d;
d->Next=q;
```

Намуди графии ин амал чунин шакл дорад:



Расми 6.

Барои дар ибтидои лист гузоштани гиреҳи аввал бо номи head ишорагаре муаррифӣ намуда, гиреҳи нав эҷод мекунем.

```
Node *head=new Node;
head->data=4;
```

Пас аз муаррифии гиреҳи нав зарурати ба ибтидои лист гузоштани он пеш меояд. Барои иҷрои ин амал бояд ишорагар (node) p ба ибтидои гиреҳи нав ишора кунад. Ба дастурҳои зерин диққат диҳед:

```
head->Next=p;
P=head;
```

Намуди графии ин амал шакли зеринро дорад: Расми 7



Расми 7.

Эҷоди листҳои якпайвандӣ

Пас аз муаррифӣ кунии навии node ба соختани худ сохтори листи пайвандӣ шуруъ мекунем. Тавре ки қайд карда будем, листи пайвандӣ аз гиреҳҳо ташкил ёфтааст, бинобар ин дар сохтори List ҳамаи амалиётҳои марбут ба листи пайвандӣ (илова, нест, ҷустуҷӯ ва ғайра) навишта мешаванд. Дар аввал ду амалиёти асосиро дар сохтори List дида мебароем. Ин амалиётҳо иборатанд аз илова намудани гиреҳи нав ва хориҷ намудани node-ҳо. Барои ба хонанда фаҳмо шудани ин матлаб намуди умумии класи List-ро дар поён меорем.

```
struct List{
    Node *head; //сар (ибтидо) и листи пайвандӣ;
    List(){ //Конструктор ;
        head=NULL; // то ҳол дар листи пайвандӣ ягон элемент нест;
    }
    // метод барои илова намудани элемент нав
    void addNode(int d){
    }
    // Метод барои хориҷ намудани элементҳои листи пайвандӣ
    void printNode() {
    }
}
```

Ин сохтор дорои як майдони ишорагар бо номи *head, як конструктор ва ду метод (функсия-аъзо) мебошад. Ишорагари *head аз навъи Node буда, ба ибтидои листи пайванди ишора мекунад. Конструктор барои ба гиреҳи нахустин бахшидани қимати NULL истифода бурда мешавад. Сабаби қимати NULL бахшидан ба тағйирёбандаи ишорагари *head дар он аст, ки ҳангоми сохтани нусхаи сохтори List он дорои ягон элемент набуда, ба ҷое ишора намекунад. Методҳои addNode ва printNode() бошанд, мувофиқан барои илова намудани элемент аз қафои листи пайвандӣ ва хориҷ намудани элементҳои листи пайванди пешбинӣ шудаанд. Акнун, ки ба сохтори листи пайвандӣ шинос шудем, матни барнома ро ба таври пурра меорем.

```
#include <iostream>
using namespace std;
struct Node
{
    int data;
    Node *Next;
};
struct List{
    Node *head; //сар (ибтидо)-и листи пайвандӣ;
    List()
    {
        head=NULL; //то ҳол элементи яқум вучуд надорад;
    }
    // метод барои илова намудани элемент нав
    void addNode(int d)
    {
        Node *nd=new Node; // ба намуди динамикӣ сохтани гиреҳи нав
        nd->data =d; // қимат бахшидан ба node;
        nd->Next= NULL;
        if (head==NULL)
            head=nd;
        else
        {
            Node *current=head;
            // дар ҳалқаи такрорӣ элементи пеш аз охири ёфта мешавад.
            while(current->Next!=NULL)
                current=current->Next;
            current->Next=nd;
        }
    }
    // Метод барои хориҷ намудани элементҳои листи пайвандӣ
    void printNode(){
        Node *current=head;
        while(current)
        {
            cout<<current->data<<endl;
            current=current->Next ;
        }
    }
}
```

```

};
int main(){
    List myList;
    myList.addNode(25);
    myList.addNode(13);
    myList.addNode(12);
    myList.addNode(3);
    myList.addNode(1);
    myList.printNode() ;
    system ("pause");
    return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

2
5
1
3
1
2
3
1

```

Барои ба хонанда фаҳмо шудан, харду методи сохтори List-ро пайдарпай мавриди баҳс қарор медиҳем.

Дар сатри якуми методи addNode() тавассути оператори new объекти (гиреҳи) нава аз навъи Node сохта шудааст. Тавре ки медонем ҳангоми иҷроиши ин сатр дар ишорагари nd суроғайи гиреҳи нав сохташуда навишта мешавад, агар сохтани node бо муваффақият анҷом гирад. Дар ҳолати акс ба он қимати NULL бахшида мешавад. Баъд аз ин тавассути дастури nd->data=d ба майдони гиреҳи сохташуда мурочиат карда шуда, ба он қимати лозимӣ навишта мешавад. Ин қимат аргументи методи addNode() мебошад. Ба ишорагар бахшидани қимати NULL маънои аз нав сохтани node-ро дорад, ки дар охири листи пайвандӣ гузошта мешавад. Ин амал тавасути дастури nd->Next= NULL иҷро карда мешавад.

Қитъаи зерини барнома муайян мекунад, ки гиреҳи нахустини листи пайвандӣ сохта мешавад ё дигар .

```

if (head==NULL)
    head=nd;
else
{
    ....
}

```

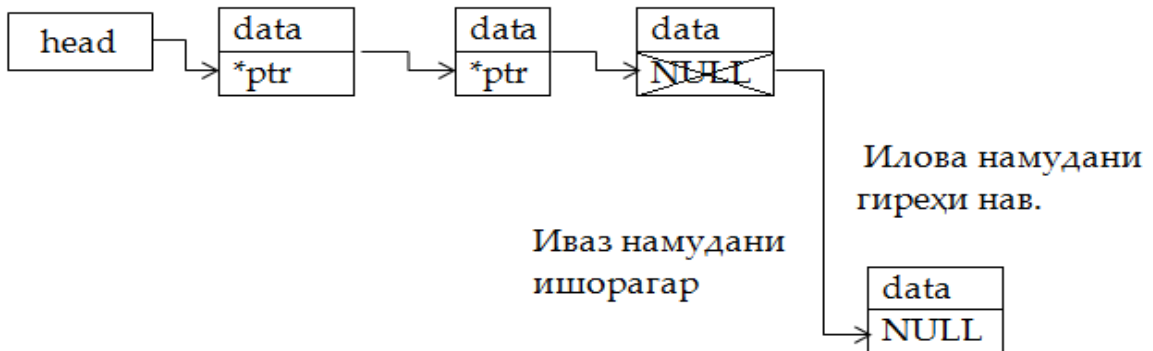
Дар сурати сохтани гиреҳи нахустин, ба ишорагари head қимати NULL бахшида мешавад ва дастурҳои пас аз шарт иҷро карда мешаванд. Дар ин сурат ба head суроғайи гиреҳи аввала аз хотира бахшида мешавад. Дар ҳолати сохтани гиреҳҳои боқимонад (2,3,4,...) қисмати дастурҳои пас аз else, ки шакли зеринро доранд иҷро карда мешаванд.

```

Node *current=head;
// дар ҳалкаи такрорӣ элементи пеш аз охири ёфта мешавад.
while(current->next!=NULL)
    current=current->next;
current->next=nd;

```

Дар ин ҷо тағйирёбандаи иловагӣ аз навъи Node муаррифӣ шуда, ба сифати қимати аввала ба он суроғи ибтидои лист бахшида мешавад. Пас аз ин дар дохили ҳалкаи такрорӣ то мавқеи гиреҳи охири сохтамон ҳаракат мекунад. Дар сурати ба он расидан суроғи гиреҳи нав сохташуда ба он бахшида мешавад. Барои пурратар дарк кардани ин матлаб, дар поён тасвири графии илова намудани гиреҳи нав дар охири лист оварда шудааст.



Расми 8.

Пас аз он, ки ба моҳияти методи addNode() пурра шинос шудем, методи printNode()-ро, ки шакли зеринро дорад мавриди баҳс қарор медиҳем.

```

void printNode() {
    Node *current=head;
    while(current)
    {
        cout<<current->data<<endl;
        current=current->Next ;
    }
}

```

Дар ин ҷо тағйирёбандаи ёрирасоне аз навъи Node сохта шуда, ба он суроғи гиреҳи якуми листи пайвандӣ бахшида шудааст. Сипас дар дохили ҳалкаи такрорӣ то расидан ба гиреҳи охири пайдарпай қиматҳои гиреҳҳо хориҷ карда мешаванд.

Тавре ки дар ин мисол дида мешавад элементи нав аз охири илова карда шудааст. Илова бар ин, равиши дигаре низ вуҷуд дорад, ки дар он элемент аз аввали лист илова карда мешавад. Барои ин зарур аст, ки барномаи қаблӣ ба сурати зерин тағйир дода шавад.

```

#include <iostream>
using namespace std;
struct Node{
    int data;
    Node *Next;
};
struct List{
    Node *head; //сар (ибтидо) и листи пайвандӣ;

```

```

List()
{
    head=NULL;    //то ҳол элементи якум вучуд надорад;
}
// метод барои илова намудани элемент нав
void addNode(int d){
Node *nd=new Node; // ба намуди динамикӣ сохтани гиреҳи нав
nd->data =d; // қимат бахшидан ба node;
// гиреҳи нав дар охир илова карда мешавад, барои ин NULL
nd->Next= head;
head=nd;
if(head==NULL) //агар гиреҳи аввал сохта шавад
    head=nd;
else // агар гиреҳи аввал набошад
{
    Node *current=head;
//дар ҳалқа гиреҳи пеш аз охири чустучӯ карда мешавад.
while (current->Next!=NULL)
    current=current->Next ;
// гиреҳи пешина ба охири ишора мекунад
current->Next =nd;
}
}
// Метод барои хориҷ намудани элементҳои листи пайвандӣ
void printNode(){
Node *current=head;
while(current!=NULL)
{
    cout<<current->data<<" ";
    current=current->Next ;
}
}
};

int main(){
List myList;
myList.addNode(25);
myList.addNode(13);
myList.addNode(12);
myList.addNode(3);
myList.addNode(1);
myList.printNode() ;
return 0;
}

```

Дар ин варианти барнома методи add Node() ба шакли зерин тағйир дода шудааст:

```

void addNode(int d) {
Node *nd=new Node; // ба намуди динамикӣ сохтани гиреҳи нав
nd->data =d; // қимат бахшидан ба node;

```

```
        nd->Next= head;
        head=nd;
    }
```

Натиҷаи барнома шакли зеринро дорад:

```
1 3 12 13 25
```

Нест намудани гиреҳ аз листи пайвандӣ

Барои нест намудани гиреҳе аз листи пайвандӣ, зарур аст, ки ишорагари онро ёфта ба NULL баробар кард. Бигзор листи пайвандие иборат аз се гиреҳ дода шуда бошад. (Расми 7). Барои нест намудани гиреҳи d кифоя аст, ки майдони Next-и ишорагари p ба q ишора кунад ва майдони Next-и ишорагари d бошад, NULL бахшида мешавад. Баъд аз ин гиреҳеро, ки ишорагари d ба он ишора мекунад, ба фазои озодӣ ҳофиза бармегардад. Ба дастурҳои зерин диққат диҳед:

```
p->Next=q;
d->Next=NULL;
delete d;
```

Нест намудани гиреҳ аз ибтидои листи пайвандӣ низ ба ҳамин монанд мебошад, вале дар ёд дошта бошед, ки ишорагаре, ки ба ибтидои лист ишора мекард, бояд ба гиреҳи баъдӣ ишора кунад. Масалан барои нест намудани гиреҳи аввалаи расми 7. чунин амал мекунем: Гиреҳи head-ро, ки ибтидои лист аст ба гиреҳи баъдӣ мебарем ва майдони Next- вобаста ба гиреҳи p ба NULL баробар карда мешавад. Дар охир p-ро ба фазои озод бармегардонем. Ба дастурҳои зерин диққат диҳед:

```
head=head->Next;
p->Next=NULL;
delete p;
```

Мисол. Функцияе нависед, ки аз листи пайвандии мисоли қаблӣ, элементи дуввумро нест кунад.

```
#include <iostream>
using namespace std;
struct Node{
    int data;
    Node *Next;
};
struct List{
    Node *head; //сар (ибтидо) и листи пайвандӣ;
    List()
    {
        head=NULL; //то ҳол элементи яқум вучуд надорад;
    }
    // метод барои илова намудани элемент нав
    void addNode(int d)
    {
        Node *nd=new Node; // ба намуди динамикӣ сохтани гиреҳи нав
        nd->data =d; // қимат бахшидан ба node;
        // гиреҳи нав дар охир илова карда мешавад, барои ин NULL
```

```

        nd->Next= head;
        head=nd;
        if(head==NULL) //агар гиреҳи аввал сохта шавад
            head=nd;
        else // агар гиреҳи аввал набошад
        {
            Node *current=head;
            //дар ҳалқа гиреҳи пеш аз охирон ҷустуҷӯ карда мешавад.
            while (current->Next!=NULL)
                current=current->Next ;
            // гиреҳи пешина ба охирон ишора мекунад
            current->Next =nd;
        }
    }
    // Метод барои хориҷ намудани элементҳои листи пайвандӣ
    void printNode(){
        Node *current=head;
        while(current!=NULL)
        {
            cout<<current->data<<endl;
            current=current->Next ;
        }
    }
    //Метод барои нест намудани элементи дуввум
    void DelSecond(){
        Node *q;
        if (head!=NULL){
            q =head->Next;
            if (q!=NULL){
                head->Next=q->Next ;
                delete q;
            }
        }
    }
};

int main(){
    List myList;
    myList.addNode(25);
    myList.addNode(13);
    myList.addNode(12);
    myList.addNode(3);
    myList.addNode(1);
    myList.printNode() ;
    cout<<endl;
    myList.DelSecond();
    myList.printNode() ;
    system ("Pause");
    return 0;
}

```


Иҷрои амалиётҳои дигар бо листҳои якпайвандии

Барои листҳои якпайвандӣ метавон амалиётҳои дигарро низ иҷро намуд.

Ҳоло дар сохтори List амалиётҳои зеринро илова мекунем:

- 1) Илова намудани гиреҳ ба ибтидои листи пайвандӣ;
- 2) Илова намудани гиреҳ дар охири листи пайвандӣ;
- 3) Рақамгузори элементҳои листи пайвандӣ;
- 4) Ҷустуҷӯи элемент дар дохили листи пайвандӣ;
- 5) Нест намудани гиреҳ аз ибтидои листи пайвандӣ;
- 6) Нест намудани гиреҳ аз охири листи пайвандӣ;
- 7) Хориҷ намудани элементҳои листи пайвандӣ аз аввал ба охир;
- 8) Хориҷ намудани элементҳои листи пайвандӣ аз охир ба аввал;

Барои иҷрои ҳар кадом ин амалиёт методи мувофиқ ба онро менависем.

Дар натиҷа мисоли қаблан овардаамон шакли зеринро мегирад:

```
#include <iostream>
using namespace std;
struct Node{
    int key;        //рақами гиреҳ дар листи пайвандӣ;
    int data;       //итилооти гиреҳ;
    Node *next;    // ишорагар ба гиреҳи баъдӣ;
    Node *last;    //ишорагар ба гиреҳи қаблӣ;
    Node(int data, Node *next, Node *last)
    {
        this->data = data;
        this->next = next;
        this->last = last;
    }
    Node(int data)
    {
        this->data = data;
    }
};

struct List{
    Node *head;    // сар (ибтидо) и листи пайвандӣ;
    Node *temp;    //ишорагар ба гиреҳи охири;
    List(){
        /* ҳангоми сохтани гиреҳи яқум ишорагар баробар бо NULL
        карда мешавад, барои он яъне гиреҳи баъди вучуд надорад;
        */
        head=NULL;
        temp=NULL;
    }
    // метод барои муайян намудани рақами гиреҳ;
    void numNode()
    {
        Node *counter = head;
        int i = 0;
        while(counter)
```

```

        {
            counter->key = ++i;
            counter = counter->next;
        }
    }
    //метод барои илова кардани гиреҳ дар озири листи пайвандӣ;
    void addNodeEnd(int data){
        //гиреҳи нав сохта онро дар охири лист қарор медиҳем;
        Node *nd = new Node(data, NULL, temp);
        temp = nd;
        if(head)
        {
            //ба ибтидои лист (гиреҳи якум) ишора мекунад
            Node *current = head;
            // Ҳалқаи такрорӣ барои муайян кардани гиреҳи баъдӣ.
            // Ин ҳалқа то расидан ба охири гиреҳ иҷро карда мешавад
            while(current->next
                current = current->next; //Гузаштан ба гиреҳи навбатӣ;
                ///гиреҳи қабли ба гиреҳи баъди ишорат мекунад;
                current->next = nd;
            }
            // агар лист холи бошад, гиреҳи нахустин сохта мешавад
            else
                head=nd;
            numNode(); //рақамгузори гиреҳ;
        }
    }
    // метод барои илова намудани гиреҳ аз аввали листи пайвандӣ
    void addNodeBeginning(int data){
        Node *nd = new Node(data);
        if(head) //агар дар лист гиреҳ алақай вучуд дошта бошад;
        {
            Node *tmp = head;
            head = nd;
            nd->next = tmp;
        }
        else //агар лист холӣ бошад;
        {
            head = nd;
            nd->next = NULL;
        }
        numNode();
    }
}
//Метод барои нест намудани гиреҳ аз аввали листи пайвандӣ;
void deleteNodeEnd(){
    if(temp)
    {
        Node *current = temp;
        current = current->last;
        delete current->next;
        current->next = NULL;
    }
}

```

```

    }
    else
        cout << "The list is empty!" << endl;
}
//метод барои нест намудани гиреҳ аз ибтидои лист;
void deleteNodeBeginning()
{
    if(head)
    {
        Node *tmp = head;
        tmp = tmp->next;
        delete head;    //озод намудани хотира аз ин суроға;
        head = tmp;    //бахшидани қимати нав;
    }
    else
        cout << "The list is empty!" << endl;
    numNode();
}
//метод барои ҷустуҷӯи гиреҳ аз дохили листи
//пайванди бо дода шудани калид;
int find(int k){
    Node *counter = head;
    while(counter)
    {
        if(counter->key == k)
            return counter->data;
        counter = counter->next;
    }
}
//Хориҷ намудани элементҳои лист бо тартиби роста;
void printListForward() const {
    Node *current = head;
    while(current)    //то расидан ба гиреҳи охирон;
    {
        //хориҷ намудани иттилоот ба саҳифаи тасвир;
        cout << current->key << " - " << current->data << endl;
        current = current->next;
    }
}
//Метод барои хориҷ намудани элементҳои листи
// бо тартиби баракс;
void printListBack() const
{
    Node *current = temp;
    while(current)
    {
        cout << current->key << " - " << current->data << endl;
        current = current->last;
    }
}
}

```

```

};
int main(){
    List myList;
    int key;
    //даъвати метод барои илова намудани гиреҳҳо аз охир
    myList.addNodeEnd(7);
    myList.addNodeEnd(69);
    myList.addNodeEnd(7);
    myList.addNodeEnd(55);
    myList.addNodeEnd(12);
    myList.addNodeEnd(99);
    myList.addNodeEnd(33);
    cout<<"Call method printListForward()"<<endl;
    myList.printListForward();
    cout << endl;
    cout<<"Call method printListBack()"<<endl;
    myList.printListBack();
    myList.addNodeBeginning(9);
    myList.addNodeBeginning(6);
    myList.addNodeBeginning(4);
    myList.addNodeBeginning(1);
    cout << endl;
    cout<<"Call method printListForward()"<<endl;
    myList.printListForward();
    myList.deleteNodeEnd();
    cout << endl;
    cout<<"Call method printListForward()"<<endl;
    myList.printListForward();
    myList.deleteNodeBeginning();
    cout << endl;
    cout<<"Call method printListForward()"<<endl;
    myList.printListForward();
    cout << "Enter a number of node: ";
    cin >> key;
    cout << "Data node " << key << " = " << myList.find(key) << endl;
    system ("Pause");
    return 0;
}

```

Дар дохили барнома, шарҳҳои лозими оварда шудаанд. Аз инру барнома хочат ба шарҳу тавзеҳ надорад.

Листҳои дупайвандӣ

Дар листҳои дупайвандӣ ҳар гиреҳи лист ба гиреҳи баъди ва қабли ишора мекунад. Бо иборати дигар ҳар гиреҳи листи дупайвандӣ суроғи гиреҳи баъдӣ ва қаблии худро дорад. Барои эҷоди чунин листҳо бояд ҳар як гиреҳи он бо ҳам пайваст карда шаванд. Аз ин ҷо дар дилхоҳ мавқеи листҳои дупайвандӣ метавон ба тарафи аввал ё охир ҳаракат кард. Ин амал барои листҳои якпайвандӣ иҷро карда намешавад. Наҳви умумии муаррификунии навъ барои листҳои дупайвандӣ (Node) шакли зеринро дорад.

```

struct Node{
    Node *left;
    Навъ итилоот ;
    Node *right;
}

```

Дар ин ҷо тавре ки дида мешавад ду ишорагар бо номҳои left ва right муаррифӣ гардидаанд. Тавре ки гуфтем, ишорагари left ба гиреҳи самти чап ва ишорагари right ба гиреҳи самти рост ишора мекунад. Итилоот метавонад дилхоҳ додаи навъи стандартӣ ва ғайристандартӣ бошад. Дар поён ба монанди мисоли қабли мисолеро дида мебароем. Барои мисоли мавриди назар ба сурати зерин Node сохта мешавад.

```

struct Node{
    Node *left;
    int data ;
    Node *right;
}

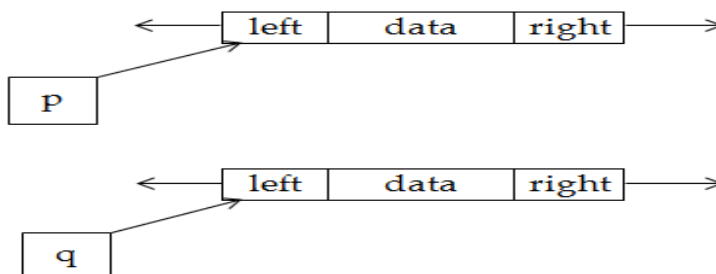
```

Пас аз муаррифукунии навъ зарурати сохтани гиреҳҳо пеш меояд. Гиреҳҳои листҳои дупайванди монанди гиреҳҳои листҳои якпайванди муаррифӣ карда мешаванд.

```
Node *p=new Node;
```

```
Node *q=new Node;
```

Бо иҷрои ин дастурҳо шакли зерин эҷод карда мешавад:



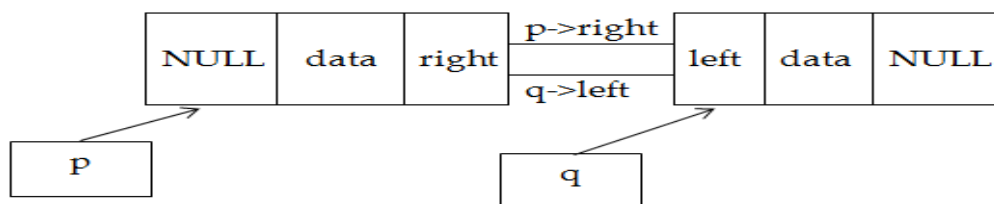
Барои пайваст кардани ин ду гиреҳ бо ҳамдигар аз дастурҳои зерин истифода мекунем:

```

p->right=q;
q->left=p;
p->left=NULL;
q->right=NULL;

```

Бо иҷрои ин дастурҳо ду гиреҳ бо ҳамдигар пайваст карда мешаванд. Намуди графии ин амал чунин аст:



Мисол. Барномае нависед, ки дар он амали сохтан ва хорич намуда-ни листи дупайванди вучуд дошта бошад.

```

#include <iostream>
using namespace std;
struct Node //сохтани навъ барои листи дупайвандӣ;
{
    Node *Prev; // ишорагар ба суроғаи гиреҳи қаблӣ;
    int x; //итилооти лист;
    Node *Next; //ишорагар ба суроғаи гиреҳи баъдӣ;
    //
};
struct List {
    Node *Head,*Tail; //ишорагар ба аввал ва охири лист;
    List(){
        Head=NULL;
        Tail=NULL ;
    }

    //Метод барои илова намудани гиреҳ дар листи дупайвандӣ;
    void Add(int x)
    {
        Node *temp=new Node; //чудо кардани хотира барои гиреҳи нав;
        temp->Next=NULL; //дар ибтидо вучуд надоштани гиреҳи баъдӣ;
        temp->x=x;//навиштани қимат дар лист;
        if (Head!=NULL) //агар лист холӣ набошад;
        {
            temp->Prev=Tail; //ба майдони мувофиқи элементти
            // баъдӣ ишора мекунем;
            Tail->Next=temp; //Суроғаи баъди ро ба қафои элемент
            //ишора мекунем
            Tail=temp; //Иваз намудани суроғаи дум;
        }
        else //агар лист холӣ бошад;
        {
            temp->Prev=NULL; //Элементи қаблӣ ба холи ишора мекунад;
            Head=Tail=temp //Сар=дум=элементе, ки ҳозир илова
            //карда шудааст;
        }
    }
}
// Метод барои хоич намудани элементҳои лист;
void Show()
{
    // Хориҷ намудани лист ах қафо
    cout<<"Print list from begin to end"<<endl;
    Node *temp=Tail;
    //Ишорагари муваққати ба суроғаи элементи охирон;
    while (temp!=NULL) //то расидан ба элементи холӣ;
    {
        cout<<temp->x<<" "; //хориҷ намудани элемент;
        temp=temp->Prev;
    }
    cout<<"\n";
}

```

```

        //хориҷ намудани лист аз аввал;
        cout<<"Print list from end to begin"<<endl;
temp=Head; //ишорагари муваққати ба суроғаи элементи нахустин;
while (temp!=NULL) //то расидан ба элементи холӣ;
{
    cout<<temp->x<<" "; //хориҷ намудани элементи хондашуда;
    temp=temp->Next; //иваз намудани суроға ба
    // суроғаи элементи баъдӣ;
}
    cout<<"\n";
}
};

int main(){
    List myList; //Муарифӣ намудани тағйирёбандаи листи дупайвандӣ;
    myList.Add(100); //Илова намудани элементҳо;
    myList.Add(200);
    myList.Add(900);
    myList.Add(888);
    myList.Show(); //Даъвати методи Show() барои хориҷ
//намудани элементҳо
    return 0;
}

```

Пуштаҳо (Stack)

Пуштаҳо додаҳои динамикие мебошанд, ки фақат онҳоро метавон аз боло дастаёбӣ кард. Дар листҳои пайвандӣ барноманавис метавонанд, ба ҳар яке аз гирехҳо дастаёбӣ дошта бошад, вале дар пушта ин тавр нест. Фарз кунед, ки мехоҳед дар сандуқе, ки андозаи васеъгии он ба як китоб баробар буда, баландиаш ба якчанд чилди китоб баробар мебошад, китобҳои худро ҷойгир кунед. Пас аз чидани китобҳо ҳар бор танҳо китоби аз ҳама болоро метавон бардошт. Барои ба даст овардани китоби аз ҳама поён зарур аст, ки ҳамаи китобҳоро аз сандуқ берун овард. Дар ин ҷо сандуқ пуштаеро мемонад, ки унсурҳои онро танҳо метавон аз боло дастаёбӣ кард.

Ба таври кулӣ, пуштаро метавон ҳамчун листи пайвандие дар назар гирифт, маҳалли дарҷ ва хазфи гирехҳо аз ибтидои он анҷом дода мешавад. Аз сабаби оне, ки ба элементи охирони пушта нисбат ба элементҳои дигар метавон зудтар дастаёбӣ шуд, бинобар ин пуштаро LIFO низ мегӯянд. LIFO муҳаффафи Last Input First Output буда, маънои “Охир омад аввал меравад”-ро дорад.

Пуштаҳоро метавон бо истифода за листҳои пайвандӣ ва массивҳо пӣдасозӣ намуд.

Бо пуштаҳо метавон амалҳои зеринро анҷом дод. Ин амалҳо иборатанд аз:

- 1) Эълони пуштаи холӣ;
- 2) Санҷидани холи будани пушта;
- 3) Илова намудани элемент дар пушта;
- 4) Нест намудани элемент аз пушта;
- 5) Ҷустуҷӯи элемент аз пушта;

ва ғайра.

Дар аввал пиёдасозии пуштаро тавассути массивҳо мавриди баҳс қарор медиҳем.

Пиёдасозии пушта бо массив

Барои пиёдасозии пуштаҳо тавассути массив, аввал массиве бо андозаи n муаррифӣ карда, сипас ду тағйирёбандаи иловагӣ барои муайянкунии ибтидо ва интиҳои пушта муаррифӣ мекунем. Масалан тарзи муаррифӣкунии пушта аз навъи бутун шакли зеринро дорад:

```
int Stack[n];  
int pos, tos;
```

Пас аз муаррифӣкунии пушта амалҳои лозимиро бо пушта ба навбат дида мебароем.

1) Эҷоди пуштаи холӣ. Барои эҷоди пуштаи холи ба тағйирёбандаҳои pos ва tos мувофиқан қиматҳои 0 ва 1-ро мебахшем.

```
pos=1;  
tos=0;
```

2) Холи будани пуштаро санҷидан.

```
if(top<pos)  
    cout<<"Stack is Empty";
```

3) Илова намудани элемент дар пушта. Барои илова намудани элемент дар пушта аввал бояд санҷида шавад, ки пушта пур нест. Ин амал ба сурати зерин иҷро карда мешавад.

```
tos++;  
if (tos>n)  
    cout<<"Stack is  
Full";  
else  
    Stack[tos]=item;
```

Қайд. Пур будани пушта, ҳангоми амалисозии он бо массив санҷида мешавад. Барои листҳои пайвандӣ ин амал вучуд надорад.

4) Нест намудани элемент аз пушта. Пеш аз нест намудани элемент аз пушта, бояд аввал санҷида шавад, ки пушта холӣ нест. Пас аз ин элементи лозимӣ аз пушта нест карда мешавад.

```
if(top<pos)  
    cout<<"Stack is Empty";  
else{  
    item=Stack[tos];  
    tos=tos-1;  
}
```

5) Бозёбии элемент аз пушта. Пеш аз иҷрои амали бозёбӣ аввал холӣ набудани пушта санҷида мешавад. Ин амал ба сурати зерин иҷро мегарданд.

```
if(top<pos)  
    cout<<"Stack is Empty";  
else  
    item=Stack[tos];
```


Пиёдасозии пушта бо листҳои пайвандӣ

Дар ин пиёдасозӣ пуштаро ба сурати листи пайвандӣ дар назар мегирем, ки элементҳо танҳо аз ибтидо илова ё нест карда мешавад.

1) Эҷоди гиреҳи пушта. Ин амл монанди листҳои пайвандӣ аст, ки myTop ба ибтидои он ишора мекунад.

2) Эҷоди пуштаи холӣ. Барои иҷрои ин амал бояд myTop баробар бо NULL карда шавад.

```
myTop=NULL;
```

3) Холӣ будани пуштаро санҷидан.

```
if(myTop==NULL)
    cout<<"Stack is Empty";
```

4) Илова намудани элемент дар пушта. Барои илова намудани элемент дар пушта аввал бояд санҷида шавад, ки пушта пур нест. Ин амал ба сурати зерин иҷро карда мешавад.

```
P= new Stack;
p->Next=myTop;
myTop=p
```

5) Нест намудани элемент аз пушта. Пеш аз нест намудани элемент аз пушта, бояд аввал санҷида шавад, ки пушта холӣ нест. Пас аз ин элементи лозимӣ аз пушта нест карда мешавад.

```
if(myTop==NULL)
    cout<<"Stack is Empty";
else
{
    h=myTop;
    item=h->data;
    myTop=myTop->Next;
    delete h;
}
```

6) Бозёбии элемент аз пушта. Пеш аз иҷрои амали бозёбӣ аввал ҳоли набудани пушта санҷида мешавад. Ин амал ба сурати зерин иҷро мегарданд.

```
if(myTop==NULL)
    cout<<"Stack is Empty";
else
    myTop=myTop->data;
```

Барои моҳияти ин амалҳоро пурра дарк кардан мисоли кучақеро дида мебароем.

```
#include <iostream>
using namespace std;
/*сохтори стек*/
struct List
{
    int x; //элементи маълумотӣ
    List *Next,*Head; //Сари пушта ва ишорагар ба элементи баъдӣ
};
/*Функсия барои илова намудани элемент ба пушта (руйхати LIFO)*/
void Add(int x, List **MyList)
```

```

{
    List *temp=new List; //Чудо кардани хойтра барои элементи нав
//дар майдони x қимате, ки аз функсия ба даст омадааст менависем
temp->x=x;
//Нишон медиҳем, ки элементи қаблӣ ин элементи баъдӣ мебошад
temp->Next>(*MyList)->Head;
//сари стекро як хонача ба тарафи рост мекуҷонем
(*MyList)->Head=temp;
}
/*Функсия барои хориҷ намудани пушта*/
void Show(List *MyList)
{
    //ишорагаре,ки ба ибтидои пушта ишора мекунад эълон мекунем
List *temp=MyList->Head;
//тавассути давр ҳамаи элементҳоро аз назар мегузаронем
while (temp!=NULL)
{
    cout<<temp->x<<" "; //хориҷ намудани элементи стек
temp=temp->Next; //гузаштан ба элементи баъдӣ
}
}
/*функсия барои нест намудани пуштаро аз ҳофиза*/
void ClearList(List *MyList)
{
    while (MyList->Head!=NULL)
    {
        List *temp=MyList->Head->Next;
delete MyList->Head;
//ивз намудани суроғаи элемент ба элементи баъдӣ
MyList->Head=temp;
}
}

int main(){
    List *MyList=new List; //Чудо намудани ҳофиза барои стек
MyList->Head=NULL; //
for (int i=0;i<10;i++)
    Add(i,&MyList); //Гузаштани элементҳо дар стек
Show(MyList); //хориҷ намудани элементҳои стек
void ClearList(List *MyList); //тоза намудани хотира.
system ("Pause");
return 0;
}

```

Мисоли дигар.

```

#include <iostream>
using namespace std;
struct stek {
    int d;
    struct stek *next; // Ишорагар ба элементи баъдӣ (стек)
};
void push(stek* &next, int d) {

```

```

// Эълон намудани тағйирёбандаи нави динамикӣ аз навъи stek
stek *pv = new stek;
pv->d = d; // бахшидани қимат
// пайваст намудани элементи навро бо элементи қаблӣ
pv->next = next;
// элементи нав ҳамчун қуллаи пушта ҳисобида мешавад
next = pv;
}
int pop(stek* &next){
// ба тағйирёбандаи temp бахшидани қимати қуллаи стек
int temp = next->d;
// барои холи намудани хотираро аз он суроғаи қуллаи пуштаро дар
//ишорагар нигоҳ медорем
stek *pv = next;
next = next->next;
delete pv;
cout<<temp; //хориҷ намудани элементи ҷорӣ
return temp; // баргардонидани қимати дар қулла буда
}

int main(){
stek *p=0;
push(p,5); //Дар стек гузоштани қимати 5
push(p,10); // Дар стек гузоштани қимати 10
pop(p); //хориҷ намудани қимати 5
pop(p); //хориҷ намудани қимати 10
system ("Pause");
return 0;
}

```

Сафҳо

Саф низ яке аз навъи сохтмони додаҳои динамикӣ аст, ки корбурди хосеро дорад. Мафҳуми саф дар барномасозӣ монанди сафи рӯзмарра аст, ки барои савор шудан ба таксӣ, харидани нон аз нонвойхона ва ғайра мебошад. Дар ин навъ сафҳо Тавре ки медонем, ҳар касе, ки зудтар ворида саф шуд, зудтар ба он хизмат расонида мешавад. Ба ҳамин далел сафро сохтмони додаҳои FIFO мегӯянд. FIFO мухаффафи First In First Out буда, маънои “Хуруҷ ба тартиби вуруд” ё “Аввал даромадан аввал хориҷ шудан”-ро дорад. Сафро низ метавон тавассути литсҳои пайвандӣ ё массивҳо пиёдаасозӣ кард. Дар ин ҷо пиёдаасозии сафро тавассути листҳои пайвандӣ дида мебароем. Ҳар саф ниёз ба ду ишорагар дорад, ки мувофиқан онҳоро front (ибтидои саф) ва rear (интиҳои саф) мегӯянд. Амали нест намудани элемент саф, аз ибтидои лист анҷом дода мешавад, ва бо ҳазфи ҳар унсуре аз саф ишорагари front ба тарафи ишорагари rear ҳаракат мекунад.

Равиши илова намудани элемент дар саф монанди илова намудани гирех дар охири листи пайвандӣ мебошад.

Нест намудани элемент аз листи пайвандӣ монанди нест намудани гирех аз интиҳои листи пайвандӣ мебошад.

Амалҳои асосие, ки бо сафҳо анҷом дода мешаванд иборатанд аз:

- 1) Эҷоди сафи ҳолӣ;
- 2) Илова намудани элемент дар саф;
- 3) Ҷустуҷӯи элемент дар дохили саф;
- 4) Нест намудани элемент аз саф;

Саволҳо барои мустаҳкамкунӣ

- 1) Листи пайвандӣ гуфта, чиро дар назар доранд?
- 2) Листи пайвандӣ чи тавр муаррифӣ карда мешавад?
- 3) Листҳои пайвандӣ чанд хел мешаванд?
- 4) Листҳои дупайвандӣ чи тавр муаррифӣ карда мешаанд?
- 5) Пушта чист ва чи фарқият аз листи пайвандӣ дорад?
- 6) Пуштаҳо чи тавр муаррифӣ карда мешаванд?
- 7) Дастеби ба элементҳои пушта чи тавр аст?
- 8) Саф чист ва чи фарқият аз пушта дорад?
- 9) Чи тавр элемент аз саф нест карда мешавад?
- 10) Оид ба листҳои пайвандӣ, пуштаҳо ва сафҳо мисол биёред.

Мисолҳо барои кори мустақилона

- 1) Функсияе нависед, ки муҳтавои элементҳои аввал ва охири листи пайвандӣ ро иваз кунад.
- 2) Барномае нависед, ки ду листи пайвандӣ ро бо ҳам муқоиса кунад.
- 3) Барномае нависед, ки қимати миёнаи арифметикии ҳамаи элементҳои листи пайвандӣ ро ҳисоб кунад.
- 4) Ду листи пайвандӣ дода шудааст. Барномае нависед, ки аз элементҳои якхелаи ин листҳои пайвандии додашуда листи пайвандии дигарро тартиб диҳад.
- 5) Барномае нависед, ки пас аз ҳамаи элементҳои токи листи пайвандӣ элементҳои додашударо гузорад.
- 6) Барномае нависед, ки амали листҳои пайвандиро таваассути массивҳо иҷро кунад.
- 7) Бо истифода аз пуштаҳо барномае нависед, ки дар мавқеаш пушида шудани қавсҳои (,) , {, }, [,]-ро муайян кунад.
- 8) Барномае нависед, ки сафҳоро дар листҳои дупайвандӣ нигоҳ дошта амалҳои лозимиро болои онҳо анҷом диҳад.
- 9) Фосилаи қимати навъҳои бутун (int, longint, shortint) маҳдуд мебошад. Масалан, қимати 15! аз ин фосила хориҷ мешавад. Барои нигоҳ доштани ададҳои калон аз листҳои дупайвандӣ истифода бурда, амалҳои зеринро анҷом диҳед:
 - а) Сумма, фарқ, нисбат ва ҳосили зарби ададҳои дароз.
 - б) Ба дараҷаи адади кутоҳ бардоштани адади дароз.
 - с) Ба дараҷаи дароз бардоштани адади дароз.
- 10) Барномае нависед, ки бо истифода аз листҳои дупайвандӣ факториали дилхоҳ адади бутуни додашударо ҳисоб кунад.

11) N адади аввалаи натуралӣ, ки аз тақсимкунандаҳои содда, танҳо 2, 3 ва 5 – ро доранд бо тартиб хориҷ карда шавад.

12) Бо истифода аз листҳои пайвандӣ барномае нависед, ки дар нимсола аз ғӯруҳи донишҷӯён, донишҷӯёни гирандаи идрорро муайян кунад. Донишҷӯ ҳамагон вақт соҳиби идрор мешавад, агар аз ҳамаи фанҳои таълими баҳои ғайриқаноатбахш гирад. Ному насаби донишҷӯ, номгӯи фанҳо ва баҳои ҳар як донишҷӯ аз ҳамаи фанҳо дода мешавад.

13) Бо истифода аз листҳои пайвандӣ ададҳои соддаи фосилаи аз 2 то n -ро хориҷ кунад.

14) Бо истифода аз листҳои пайвандӣ ададҳои мукаммали фосилаи аз 1 то n -ро хориҷ кунад.

15) Барномае нависед, ки элементҳои аввалаи ҷуфти листи пайвандиро нест кунад.

16) Барномае нависед, ки пас аз элементҳои дуввуми листи пайвандӣ элементҳои додашударо гузорад.

17) Барномае нависед, ки пас аз ҳар элементҳои манфӣ, элементҳои додашударо гузорад.

18) Барномае нависед, ки гиреҳи n -уми листи пайвандиро нест кунад.

19) Барномае нависед, ки элементҳои соддаи листи пайвандиро нест кунад.

20) Барномае нависед, ки гиреҳи n -уми листи пайвандиро ба квадрат бардорад.

Боби 12. Китобхонаи қолаб (шаблон)-ҳои стандартӣ STL (Standard Template Library)

Муқаддима

Китобхонаи STL дар забони C++ яке аз китобхонаҳои нисбатан бузурге ба ҳисоб меравад, ки дар он функцияҳои зиёде барои кор кардан бо листҳо, векторҳо, алгоритмҳо, сатрҳо ва ғайра мавҷуданд. Ба таври дигар гӯем, STL –ин маҷмӯи қолаб (шаблон)-ҳои class-ҳо ва функция(function)-ҳо мебошад. Агар ҳамаи STL-ро гирд оварем, як китоби мураттабе ҳосил мушавад. Аз ин рӯ, баъзан қисматҳои муҳими онро меорему халос. Воқеан STL-таъминоти барномавии аз ҳама васеъ дар забони C++ ба ҳисоб меравад. Аз ин рӯ, лозим донистем, ки дар ин китоб танҳо як қисмати ниҳоят хурди ин библиотекро мавриди баррасӣ қарор дода, қисмати дигарашро ҳавола бар хонанда созем. Тавре ки қайд намудем, STL-қисмати мураккабтарин нисбат ба қисматҳои дигар мебошад ва агар хонандаи арҷманди мо ҳамаи маводи бобҳои гузаштаро аз худ карда бошад, метавонад ба осони ин бобро низ аз худ кунанд. Дар ин боб аслан аз қолабҳо истифода бурда мешавад. Шояд баъзе хонандагони мо суол диҳанд: зарурати истифодабарии қолабҳо дар чист, оё функцияҳо наметавонанд ҳамаи талаботи моро коней гардонанд? Барои ба ин савол ҷавоб гардонидан як мисоли кучакеро дида мебароем.

Мисоли 1. Функцияе нависед, ки аз ду адади додашуда калонта-ринашро муайян кунад.

```
#include <iostream>
using namespace std;
int max(int a, int b)
{
    return a>b?a:b;
}

int main()
{
    int a, b;
    cin>>a>>b;
    cout<<max(a,b);
    return 0;
}
```

Тавре ки дида мешавад, дар ин мисол функцияи $\max(a, b)$ аз ду адади бутуни додашуда калонтаринашро бармегардонад. Акнун фарз мекунем, ки аргументҳои мо ҳақиқӣ ё рамзӣ бошанд, дар ин сурат функцияи мазкур ба хатогӣ рубарӯ мешавад, зеро он танҳо барои ададҳои бутун навишта шудааст. Пас, барои он ки функцияи мо барои ҳамаи навъҳои додашуда кор кунад, зарур аст, ки барои ҳар як навъ онро аз сари нав нависем. Ин амал коди барномавро тулонӣ гардонидани, суръати иҷроиши онро суст мегардонад. Илова бар ин, ислоҳи хатогӣҳо низ душвор мегардад. Пас, барои аз ин душворӣ раҳо шудан функцияи лозимиро як маротиба бо истифода аз шаблонҳо навишта барои ҳамаи навъҳо истифода мебарем. Биёед мисоли 1-ро аз сари нав бо истифода аз қолабҳо менависем:

```

#include <iostream>
using namespace std;
template <class T>
T max( T a, T b){
    return a>b?a:b;
}
int main(int argc, char* argv[]){
    int a, b;
    double x, y;
    char c, c1;
    cout<<"du adadi butun dohil kuned:\n";
    cin>>a>>b;
    cout<<"max("<<a<<","<<b<<")="<<max(a,b);
    cout<<"\ndu adadi hakiki dohil kuned:\n";
    cin>>x>>y;
    cout<<"max("<<x<<","<<y<<")="<<max(x,y);
    cout<<"\n du simbol dohil kuned:\n";
    cin>>c>>c1;
    cout<<"max("<<c<<","<<c1<<")="<<max(c,c1);
    return 0;
}

```

Дар ин барнома функсияи `max` ба намуди қолаб муаррифӣ гардидааст. Дар функсияи асосӣ ҳангоми даъвати ин функсия тағйирёбандаҳои навъҳои гуногун ба сифати аргументи он иштирок мекунанд ва аз натиҷа дида мешавад, ки функсияи мазкур рӯи дилхоҳ навъи додашуда кор мекунад.

Назардошти умуми дар бораи библиотекаи STL

Элементҳои асосии STL, ки ядрои он низ номида мешаванд, иборатанд аз:

1. Контейнерҳо
2. Алгоритмҳо
3. Итераторҳо

Контейнерҳо-ин объектҳои мебошанд, ки барои объекти дигарро дар худ нигоҳ доштан хизмат мекунанд. Дар STL якчанд навъи контейнерҳо мавҷуданд, ба монанди: векторҳо, листҳо, қаторҳои дутарафа, ва ғайра. Класси `vector` – барои кор кардан ба масивҳои пӯё (динамикӣ), класси `list`- барои кор кардан бо рӯйхатҳо, класси `queue` –барои сохтани қаторҳои дутарафа ва ғайра хизмат мекунанд. Ба ғайр аз инҳо, дар STL контейнерҳои ассотсиативӣ мавҷуданд, ки ҳар кадомаш қиматеро аз рӯи калиди додашуда соҳиб мешавад. Ба ин мисол шуда метавонад класси `map`.

Ҳар як класси контейнер маҷмӯи функсияҳоеро истифода мекунад, ки ба худаш хосанд. Масалан, контейнери `list` функсияҳоеро барои гузоштан, нест кардан, якҷоя кардан ва ғайра истифода мекунад.

Алгоритмҳо-барои ба контейнерҳо бахшидани қимати аввала, мурағаттаб намудани элементҳои контейнер, ҷустуҷӯ намудани элементе аз дохили контейнер, иваз намудани ягон элементи контейнер ва ғайра истифода истифода бурда мешаванд.

Итераторҳо –объектҳои мебошанд, ки ба монанди ишорагарҳо амал мекунад. Онҳо бо ёрии калимаи калидии `iterator` дар контейнерҳои гуногун муаррифӣ карда мешаванд. Вазифаи асосии итераторҳо барои суроғаи контейнерҳо дар худ нигоҳ доштан мебошад.

Дар STL на танҳо контейнерҳо, алгоритмҳо ва итераторҳо мавҷуданд, балки компонентҳои стандартии дигаре низ мавҷуданд, ки баъдтар мавриди баҳс қарор мегиранд.

Ҳар як контейнер ба худ ҷудокунии ҳофиза (`allocator`)-ро дорад. Ҷудокунии ҳофиза –ин ҳофизаи барои контейнер ҷудошударо идора (`control`) мекунад. Ҷудокунии стандартӣ -ин объекти классии `allocator` мебошад ва дар вақти зарурӣ метавон аз ҷудокунии махсус низ истифода кард.

Якчандто аз контейнерҳо ва алгоритмҳо навъи махсусро, ки предикат (`predicate`) ном дорад, истифода мебаранд. Аслан ду намуди предикатҳо вучуд дорад ягона ва дугона. Предикатҳои ягона дорои як аргумент буда, тавассути калимаи калидии `UnaryPredicate` муаррифӣ карда мешаванд. Предикатҳои дугона бошанд, дорои ду аргумент буда, тавассути калимаи калидии `BinaryPredicate` муаррифӣ карда мешаванд.

Илова бар ин, `class`-ҳои гуногуни STL бо пайваст кардани библиотекҳои стандартии C++ ба қор мебароянд. Ин библиотекаҳо `<utility>` ва `<functional>` мебошанд. Дар библиотекаи `<utility>` шаблони классии `pair` муаррифӣ карда шудааст, ки он ҷуфти қиматҳо, ки калид ва қимат ном доранд, дар худ нигоҳ медоранд. Дар библиотекаи `<functional>` бошад, қолабҳои муаррифӣ шудаанд, ки барои сохтани объектҳои, ки онҳо функсияи `operator()` муайян мекунад, истифода бурда мешаванд. Ин объектҳои функционалӣ номида шуда, дар алгоритмҳо ба сифати ишорагар истифода карда мешаванд.

Объектҳои функционалӣ

Объектҳои функционалӣ классе мебошанд, ки дар он амали даъвати функсия муайян карда шудааст. Одатан ин хел объектҳо ба сифати параметри алгоритмҳои стандартӣ истифода бурда мешаванд. Тавассути ин объектҳои истифодабаранда метавонад, барои муқоисаи объектҳо ва қоркарди онҳо меъёре (критерияе) эҷод кунад.

Барои он алгоритмҳои, ки объектҳои функционалӣ истифода бурда мешаванд, метавон ишорагар ба функсияро низ истифода кард. Истифодаи объектҳои функционалӣ назар ба ишорагари функсия пуртаъсиртар мебошад, зеро оператори `()` метавонад, ҳамчун оператори дохилӣ муаррифӣ гардад.

Объектҳои функционалӣ дар библиотекаи `<functional>` ҷойгир мебошанд. Аз байни ин объектҳо, метавон он объектҳои, ки қимати мантиқӣ бармегардонанд, предикат номида мешаванд. Функсияҳои оддӣ, ки қимати мантиқӣ бармегардонанд, низ предикат мебошанд.

Дар STL қолабҳои унарӣ ва бинарии объектҳои функционалӣ муайян гардидааст.

```
template <class Arg, class Result>
struct unary_function{
    typedef Arg argument type;
```



```

typedef Result result_type;
};
template <class Arg1, class Arg2, class Result>
struct binary_function{
    typedef Arg1 first_argument_type;
    typedef Arg2 second_argument_type;
    typedef Result result_type;
}

```

Объектҳои функционалии STL зодаи ин функция-объектҳои базавӣ мебошанд. Онҳо одатан якҷоя бо адаптерҳо ва дигар воситаҳои забон истифода бурда мешаванд.

Дар STL якчанд навъи адаптерҳо мавҷуданд:

- ✓ Пайваस्तкунандаҳо-барои истифодабарии объектҳои функционалии дуаргумента ҳамчун объекти якаргумента;
- ✓ Инкоркунандаҳо -барои ивазкунии қимати предикат;
- ✓ Адаптерҳои ишорагар ба функция;
- ✓ Адаптерҳои методӣ - барои истифодабарии методҳо дар алгоритмҳо.

Объектҳои функционалии арифметикӣ

Дар STL қолабҳои объектҳои функционалӣ, барои ҳамаи операторҳои арифметикӣ, муайян гардидааст.

Ном	Навъ	натича
plus	бинарӣ	x+y
minus	бинарӣ	x-y
multiplies	бинарӣ	x*y
divides	бинарӣ	x/y
modulus	бинарӣ	x%y
negate	унарӣ	-x

Дар поён қолаби объекти plus оварда шудааст. (Объектҳои дигар низ ба он монанд мебошанд.)

```

template <class T> struct plus: binary_fuction<T,T,T>{
    T operator () (const T& x, const t& y) const{
        return x+y;
    }
};

```

Аз сабаби он ки объектҳои функционалӣ якҷоя бо алгоритмҳо истифода бурда мешаванд, мисоли онҳоро ҳангоми омӯхтани алгоритмҳо дида мебароем.

Предикатҳо

Дар STL қолабҳои объектҳои функционалӣ барои муқоиса ва иҷрои амалҳои мантиқӣ муайян гардидаанд, ки қимати мантиқӣ бармегардонанд, яъне онҳо предикат мебошанд.

Ном	Навъ	Натиҷа
equal_to	бинарӣ	x==y

not_equal_to	бинарӣ	x!=y
Greater	бинарӣ	x>y
Less	бинарӣ	x<y
greater_equal	бинарӣ	x>=y
less_equal	бинарӣ	x<=y
logical_and	бинарӣ	x&&у
logical_or	бинарӣ	x у
logical_not	унарӣ	!x

Дар поён қолаби объекти equal_to оварда шудааст. Объектҳои дигар бо он монанд мебошанд.

```
template <class T> struct equal_to:
    binary_function<T, T, bool>{
        bool operator()(const T& x, const T& y) const{
            return x==y;
        }
    };
```

Илова бар ин барномасоз метавонад, барои худ предикат нависад. Ин амал он вақт лозим мешавад, ки агар элементҳои контейнер аз навъи истифодабаранда иборат бошанд. **Мисол:**

```
struct my_struct:
    public binary_function<myst , myst, bool>{
        bool operator () (myst& ms, myst& ms){
            return m1.get_myst ()<m2.get_myst ();
        }
    };
```

Инкоркунандаҳо

Инкоркунандаҳои not1 ва not2 барои ба даст овардани инверсияи предикатҳои бинарӣ ва унарӣ истифода бурда мешаванд. Истифодаи инкоркунандаҳо хело содда мебошад, масалан барои ба даст овардани инверсияи equal_to<int>() дастури not2(equal_to<int>()) кифоягӣ мекунад. Ин дастур бо not_equal_to<int>() баробарқувва мебошад.

Асосан инкоркунандаҳо барои инверсияи предикатҳои сохтаи барномасоз пешбинӣ шудаанд. Барои предикатҳои STL бошад, инверсияҳои тайёр мавҷуд аст.

Татбиқи инкоркунандаҳо ро дида мебароем. Дар STL объектҳои unari_negate ва binary_negate, ки мувофиқан зодаи unary_function ва binary_function мебошанд, мавҷуданд. Ин классҳо ба сифати параметр конструктори объектеро ки предикати навъи матлуб мебошад, қабул мекунанд.

```
template <class Predicate>
class unari_negate: public
    unary_function<typename Predicate:: argument_type, bool>{
public:
    explicit unari_negate(const Pradicate &pred);
    bool operator()
```

```
(const typename Predicate::argument_type &x)const;
};
```

Амалгари () инверсияи предикати not1 ва not2-ро ҳамчун шаблони функсия бармегардонад. Чунин функсия предикат қабул карда, объекти функционалӣ бармегардонад.

```
template <class Predicate>
    unary_negate<Predicate not1(const Predicate &pred);
template <class Predicate>
    binary_negate<Predicate not1(const Predicate &pred);
```

Пайвасткунандаҳо

Бо ёрии предикатҳои бинарӣ метавон, ду объектҳои гуногунро муқоиса кард. Баъзан зарурати муқоисаи объект бо доимӣ пеш меояд. Ҳангоми истифодаи предикатҳо, барои ин мақсад, лозим меояд, ки аргументҳои он бо доимӣ пайваст карда шаванд. Барои иҷрои ин амал аз пайвасткунандаҳои bind2nd ва bind1st истифода бурда мешавад. Ин пайвасткунандаҳо, мувофиқан, барои ба қимати аниқе пайваст кардани аргументи дуввум ва яқуми функсияи бинарӣ истифода бурда мешаванд.

Пайвасткунандаҳо дар STL ҳамчун қолаби функсия истифода бурда мешаванд, ки ба сифати аргументи аввала, объекти функционалии дорой ду аргументро қабул карда, ба сифати аргументи дуввум бошад, алоқакунандаи қимати value-ро истифода мекунад.

Барои муайян кардани навъе, ки объекти функционалӣ бармегардонад, дар STL қолаби классҳои binder2nd ва binder1st тавзеҳ дода шудааст. Қолаби ин пайвасткунандаҳо шакли зеринро дорад:

```
template <class Op, class T>
    binder2nd<op> binder2nd(const Op &op, const T &x)
template <class Op, class T>
    binder1st<op> binder1st(const Op &op, const T &x)
```

дар инҷо Op-навъи объекти функционалӣ буда, T бошад, навъи алоқакунандаи қимат мебошад.

Мисол. Ҳисоб намудани миқдори элементҳои ба 20 баробари массив.

```
#include <iostream>
#include <functional.h>
#include <algorithm.h>
using namespace std;
int main()
{
    int x[10]={5,20,33,56,20,23,20,17,18, 33};
    cout<<count_if(x,x+10,bind1st(equal_to<int>()),20));
    return 0;
}
```

Дар инҷо барои ҳисоб намудани миқдори элементҳо аз алгоритми count_if() истифода бурда шудааст, ки дар оянда дар ин бора тавзеҳ хоҳем дод. Ду аргументҳо аввалаи ин алгоритм итератор буда, мувофиқан аввал ва охири

пайдарпайиро нишон медиҳанд. Аргументи сеюми бошад, функсияи бинари буда, шартӣ додашудаи $x==y$, яъне $x=20$ -ро месанҷад.

Адаптерҳои ишорагар ба функсия

Барои пайваст кардани ишорагарҳои оддиро ба функсия, аз табдилдиҳии махсус ва ё адаптерҳо истифода бурда мешавад. Дар STL ду объекти функционалӣ: ишорагар ба функсияи унарӣ `pointer_to_unary_function` ва ишорагар ба функсияи бинарӣ `pointer_to_binary_function` ва инчунин ду функсия-адаптер `ptr_fun` бо як ва ду аргумент муаррифӣ гардидаанд. Ин функсия-адаптерҳо ишорагар ба функсияи ба сифати аргументашон фиристодашударо ба объекти функционалӣ табдил медиҳанд.

```
template <class Arg, class Result>
    class pointer_to_unary_function:public unary_function<Arg, Result>{
    public:
        explicit pointer_to_unary_function(Result (*f) (Arg));
        Result operator () (Arg x) const;
    };
template <class Arg, class Result>
    pointer_to_unary_function <Arg, Result>
        ptr_func(Result (*f) (Arg));
        result operator () (Arg x) const;
template <class Arg1, class Arg1, class Result>
    class pointer_to_binary_function:public
        binary_function<Arg1, Arg1, Result>{
    Public:
        explicit pointer_to_binary_function(Result (*f) (Arg1, Arg2));
        Result operator () (Arg x, Arg2 y) const;
    };
template <class Arg1, class Arg1, class Result>
    pointer_to_binary_function<Arg1, Arg1, Result>{
        ptr_func(Result (*f) (Arg1, Arg2));
```

Мисол. Истифодаи адаптери функсия.

```
#include <iostream>
#include <functional.h>
#include <algorithm.h>
using namespace std;

struct MyStruct{
    int a, b;
};

bool lss( MyStruct x, MyStruct y){
    return x.a==y.a;
}

int main()
```

```

{
    MyStruct x[10]={{1,5},{2,3},{6,1},{5,6},{3,5},{2,3},
                  {2,9},{1,3},{1,3}, {3,7}};
    MyStruct e1={3,4};
    cout<<count_if(x,x+10,bind2nd(ptr_fun(lss),e1));
    return 0;
}

```

Алгоритми стандартии `count_if` миқдори элементҳое ро, ки шарти аргументи сеюмро қаноат мекунад, ҳисоб менамояд. Ин параметр объекти функционалӣ буда, онро алоқақунандаи `bind2nd` бо ёрии адаптери `ptr_fun` аз функсияи `lss` ба даст овардааст. Натиҷаи барнома ба 2 баробар мебошад.

Адаптери методҳо

Ҳангоми дар контейнер нигоҳ доштани объектҳои классии истифодабаранда, баъзан зарурати татбиқи як методи класс ба ҳамаи элементҳои контейнер пеш меояд. Барои аз назар гузаронидани элементҳои контейнер, дар библиотекаи стандартӣ алгоритми `for_each` мавҷуд аст. Ба сифати аргументи алгоритми мазкур, ишорагари функсия равон карда мешавад. Ишорагари равонкардашуда, барои ҳамаи элементҳои контейнер алоҳида-алоҳида даъват карда мешавад.

```

#include <iostream>
#include <functional>
#include <algorithm>
using namespace std;

void print(int x){
    cout<<x<<" "<<endl;
}

int main(int argc, char* argv[]){
    int x[10]={5,20,33,56,20,23,20,17,18, 33};
    for_each(x,x+10, print);
    return 0;
}

```

Адаптери методҳо, ба сифати аргументи алгоритмҳои стандартӣ имконияти истифодаи методҳоро фароҳам меоранд. Монанди пайвасткунандаҳо ва инкоркунандаҳо онҳо низ функсияро ба даст оварда аз он функсияи дигарро месозанд. Дар STL якчандто адаптерҳо, бо миқдори гуногуни аргументҳо, барои методҳо муайян гардидаанд. Дар чадвали зерин ин адаптерҳо гирд оварда шудаанд.

Ном	Навъи объект	Маъно ва мафҳум
<code>mem_fun</code>	<code>mem_fun_t</code>	Методи беаргументро тавасути ишорагар даъват мекунад.
<code>mem_fun</code>	<code>const_mem_fun_t</code>	Методи доимии беаргументро тавасути ишорагар даъват мекунад.
<code>mem_fun</code>	<code>mem_fun1_t</code>	Методи унарро тавасути ишорагар даъват мекунад.
<code>mem_fun_ref</code>	<code>mem_fun_ref_t</code>	Методи беаргументро тавасути

		нишонагар даъват мекунад.
mem_fun_ref	const_mem_fun_ref_t	Методи беаргументи доими тавассути нисонагар даъват мекунад.
mem_fun_ref	mem_funl_ref_t	Методи унариро тавассути нисонагар даъват мекунад.
mem_fun_ref	const_mem_funl_ref_t	Методи унарии доими тавассути нисонагар даъват мекунад.
mem_fun_ref	const_mem_funl_t	Методи унарии доими тавассути ишорагар даъват мекунад.

Контейнерҳои class-ҳо

Тавре ки дар гузашта қайд намудем, контейнерҳои объектҳои STL буда, барои нигоҳ доштани додаҳо дар худ истифода бурда мешавад. Дар ҷадвали зерин якҷандто аз контейнерҳои гирд оварда шудаанд.

№	Контейнер	Маъно ва мафҳум	Библиотека
1	bitset	Маҷмӯи бити	<bitset>
2	deque	Дек (қаторҳои дусамта)	<deque>
3	list	Рӯйхатҳо	<list>
4	map	Тасвирҳо	<map>
5	multimap	Тасвирҳои васеъкардашуда	<map>
6	multiset	Маҷмӯҳои васеъкардашуда	<set>
7	queue	Навбатҳо	<queue>
8	set	Маҷмӯҳо	<set>
9	vector	Қор ё векторҳо	<vector>

Аз сабаби он ки номи навъ дар мурификунии шаблонҳои class тавассути typedef-и типҳои муайян мегардад. Дар ҷадвали зерин якҷандто аз typedef-ҳои машҳур барои намуна оварда шудаанд.

№	Typedef	Маъно ва мафҳум
1	size_type	Навъи бутун ба size_t монанд мешавад.
2	reference	Нишона ба элемент
3	const_reference	Нишонаи доими ба элемент
4	iterator	Итератор
5	const_iterator	Итераторҳои доимӣ
6	reverse_iterator	Итераторҳои реверсӣ
7	const_reverse_iterator	Итераторҳои реверсии доими
8	value_type	Навъи қимате, ки дар контейнер нигоҳ дошта мешавад
9	allocator_type	Навъи тақсимшуда
10	key_type	Навъи калид
11	key_compare	Навъи функсияе, ки ду калидро муқоиса мекунад
12	mapped_type	Навъи қимате, ки дар тасвирҳои сабт мешавад

13	value_compare	Навъи функцияе, ки ду қиматро муқоиса мекунад
----	---------------	---

Векторҳо

Яке аз контейнерҳое, ки нисбатан истифодаи васеътар доранд, ин векторҳо мебошанд. Класси вектор, тавре ки қаблан қайд карда будем, барои кор кардан бо массивҳои пӯё пешбинӣ шудааст. Массиви динамикӣ массивест, ки андозаи онро метавонем дар ҳолатҳои лозими зиёд ё кам кунем. Андозаи массивҳои муқаррарӣ ё статикӣ як адади доимӣ буда, пас аз тарҷума (compile) шудани барнома наметавонистем, ки андозаи онро тағйир диҳем. Ҳалли ин масъала дар забони C++ бо ду усул пешбинӣ шудааст, яке аз ин усулҳо истифода аз ишорагар ба номи массив буд. Усули дигар, ки нисбатан васеътар аст, векторҳо мебошанд. Наҳви умумии муаррификунии объектҳои класси vector ба сурати зерин аст:

```
template <class T, class Allnameor=Allnameor<T>>
```

Дар ин ҷо T тип ё навъест, ки додаҳо бо он навъ дар ҳофиза захира карда мешаванд. Allnameor бошад, барои додаҳо ҳофизаеро ҷудо ё муайян мекунад.

Класси вектор дорои конструкторҳои стандартии зерин мебошад:

- 1) explicit vector (const allnameor &a=Allnameor());
- 2) explicit vector (size_type num, const T &Val=T(), const Allnameor &A=allnameor ());
- 3) vector (const vector <T, Allnameor> &ob);
- 4) template <class InIter> vector (InIter start, InIter end, const allnameor &a=Allnameor());

Конструктори якум барои сохтани вектори ҳолӣ пешбини шудааст. Конструктори дуюм бошад, барои сохтани векторе, ки дорои num элемент буда, ҳар элементи он дорои қимати val мебошад, пешбинӣ шудааст. Конструктори сеюм наҳви сохтани вектореро ифода мекунад, ки элементҳои он аз вектори ab гирифта мешаванд. Конструктори охири он бошад, барои сохтани векторе, ки элементҳои он дар фосилаи аз start то end меҳобанд, истифода бурда мешавад. Ҳоло тарзи муаррифӣ намудани як чанд векторро меорем.

- 1) vector <double> dv; //сохтани вектор аз ҳақиқӣ бутун бо андозаи 0
- 2) vector <int> iv; //сохтани вектори ҳолӣ аз навъи бутун
- 3) vector <char> chr; //сохтани вектори ҳолӣ аз навъи символӣ
- 4) vector <char> chr(5,'a'); // сохтани вектор бо қимати аввалаи панҷто а
- 5) vector <int> ivt(iv); //сохтани вектор дар асоси вектори мавҷудбуда

Барои объектҳои класси вектор операторҳои муқоисакунии зе-рин муайян карда шудаанд.

==, <, <=, >, >=, !,

Ба ғайр аз ин, барои класси вектор оператори “[]” муайян гардидааст, ки он барои дастрас шудан ба элементҳои вектор пешбинӣ шудааст. Илова бар ин, дар класси вектор функцияҳои зиёде муаррифӣ шудаанд, ки барои кор кардан бо векторҳо хизмат мекунанд. Барои ба ин функцияҳо дастрас шудан, яъне истифода кардан аз векторҳо бояд, ки библиотекаи <vector.h> ба барнома

пайваст карда шавад. Дар чадвали зерин якчандто аз ин функцияҳои кор бо векторҳо оварда шудаанд.

№	Функция (Функция -член)	Маъно ва мафҳум
1	template <class InIter> void assign (InIter Start, inIter end)	Пайдарпаие, ки дар фосилаи аз start то end муайяншударо дар вектор мегузорад.
2	void assign (size_type num, const T &val)	Дар вектор num-то элемент бо қимати дар параметри val нишондодашуда мегузорад
3	reference at (size_type i); const_reference at (size_type i) const;	Нишонагар ба элементе, ки дар параметри I нишон дода шудааст, бармегардонад.
4	reference back(); const_reference back() const;	Суроғи элементи охирони векторро бармегардонад.
5	iterator begin(); const_iterator begin() const;	Барои элементи якуми вектор итератор бармегардонад.
6	size_type capacity() const;	Андозаи ҷорӣ векторро бармегардонад
7	void clear();	Ҳамаи элементҳои векторро нест мекунад.
8	bool empty() const;	Агар вектор элемент надошта бошад, қимати ҳақ бармегардо-над.
9	iterator end(); const_iterator end() const;	Барои элементи охирони вектор итератор бармегардонад.
10	iterator erase (Iterator start, iterator end);	Элементҳое, ки дар фосилаи аз start то end нишон дода шудаанд аз вектор нест мекунад.
11	reference front(); const reference front() const;	Барои элементи якум дар вектор нишонагар бармегардонад
12	allnameor_type get allnameor () const;	Ҳофизаи барои вектор чудо-шударо бармегардонад.
13	iterator insert (Iterator I, const T & val=T());	Пеш аз элементе, ки дар параметри I нишон дода шуда-аст, қимати val-ро мегузорад.
14	Template <class InIter> void insert (Iterator I, InIter start, InIter end);	Пеш аз элементе, ки дар пара-метри I нишон дода шудааст, пайдарпайии элементҳои аз start то end -ро мегузорад.
15	size_type max_size() const	Барои муайян намудани эле-менти максималии вектор хизмат мекунад .
16	void pop_ back()	Элементи охирони векторро нест мекунад.
17	void pop_ back(const T &val)	Элементе, ки дар параметри val нишон дода шудааст, дар охири вектор

		мегузорад.
18	reference Iterator rbegin(); const reference_iterator rbegin () const	Итератори реверсивӣ ба охири вектор бармегардонад.
19	void resize (size_type num)	Андозаи векторро баробари num мекунад.
20	void resize (size_type num T val=T())	Андозаи векторро баробари num мекунад.. Агар вектор аз num камтар элемент дошта бошад, пас элементе, ки дар параметри val нишон дода шудааст, дар охири вектор ҷойгир карда мешавад.
21	size_type size() const;	Миқдори ҷории векторро бармегардонад.
22	void swap(deque<T,Allnameor> &ob);	Элементҳои вектори даъват-шударо ба элементҳои ob иваз мекунад.

Тавре ки қаблан ишора карда будем, итераторҳо низ монанди ишорагарҳо мебошанд. Пас, функсияи begin() итератореро, ки ба аввали вектор ишора мекунад, бармегардонад. Ба ҳамин монанд функсияи end () низ итератореро, ки ба охири вектор ишора карда шудааст, бармегардонад.

Функсияи push_back() қимат ё қиматҳои додашударо дар охири вектор илова мекунад. Дар ин ҳолат, андозаи вектор ба таври автоматӣ тағйир меёбад. Бо ёрии функсияи insert() элементҳои лозимиро метавонем дар дилхоҳ ҷойи вектор илова намоем ва тавассути функсияи erase () бошад, метавонем элемент ё элементҳои нолозимиро аз векторро нест кунем. Функсияи size() бошад, андозаи векторро муаян мекунад ва ба ҳамин монанд функсияҳои дигар низ амалеро болои векторҳо иҷро мекунанд. Як чизро набояд фаромӯш кард, ки ҳамаи функсияҳои дар ҷадвали боло овардашуда, функсия-аъзо мебошанд ва якҷоя бо номи вектор истифода бурда мешаванд.

Мисол. Тарзи муаррифӣ намудани вектор ва истифода функсияҳои size() ва push_back() оварда шудааст.

```
#include <iostream>
#include <functional.h>
#include <algorithm.h>
using namespace std;
#define n 20
int main(int argc, char* argv[])
{
    vector <int> v;
    int i;
    for (i=2;i<=n;i+=2)
        v.push_back(i) ;
    cout<<"Andozai vector="<<v.size()<<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" « ;
```

```

    cout<<endl;
    for (i=1;i<=n ;i+=2)
        v.push_back(i) ;
    cout<<"Andozai navi vector="<<v.size() <<endl;
    for (i=0;i<v.size();i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    return 0;
}

```

Дар ин мисол, он тавре, ки мушоҳида мешавад, дар аввал вектори холие бо номи *v* муаррифӣ карда шуда, сипас бо он даҳ адади чуфт бахшида шудааст, дар натиҷаи ин амал андозаи вектор баробар ба 10 мешавад. Баъдан 10 адади тоқи дигар ба охири вектор илова карда мешавад ва андозаи нави вектор баробар ба 20 мешавад. Натиҷаи барнома намуди зеринро дорад:

```

Andozai vector=10
2 4 6 8 10 12 14 16 18 20
Andozai navi vector=20
2 4 6 8 10 12 14 16 18 20 1 3 5 7 9 11 13 15 17 19

```

Массивҳо ба сифати элемент дар вектор

Тавре ки гуфтем, вектор ин массиви навъи пуё (*dynamic*) мебошад. Пас, элементҳои он метавонанд элементҳои як массиве, ки қаблан муаррифӣ шудааст, бошанд.

Мисол. Барномае нависед, ки дар он ба сифати элементҳои вектор массиве иштирок кунад.

```

#include <iostream>
#include <functional.h>
#include <algorithm.h>
using namespace std;
#define n 20
int main(int argc, char* argv[])
{
    vector <int> v;
    int a[n];
    int i;
    for (i=0;i<n;i++)
        a[i]=random(25)+1;
    for (i=0;i<n;i++)
        v.push_back(a[i]) ;
    cout<<"Andozai vector="<<v.size() <<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    return 0;
}

```

Тавре ки дар ин ҷо айён аст, дар аввал массиве бо номи *a* ки дорои *n* элемент мебошад, муаррифӣ шудааст, сипас, элементҳои ин массив аз ададҳои тасодуфӣ пур карда шудаанд. Дар дохили ҳалқа элементҳои массив ба вектор

илова карда мешаванд. Барои элементҳом массивро ба вектор илова кардан, бояд массив ва вектор аз як навъ бошанд, вагарана барнома ба хатоғӣ рӯбарӯ мешавад. Натиҷаи барнома намуди зеринро дорад:

```
Andozai vector=20
25 5 12 20 13 9 7 10 12 12 20 1 8 19 5 14 24 7 21
```

Дастрасшавӣ ба элементҳои вектор тавассути итераторҳо

Ба элементҳои вектор метавон тавассути итераторҳо низ дастрас шуд. Ин амалро мо дар гузашта, тавассути ишорагарҳо дида баромада будем. Яъне дар гузашта дида будем, ки элементҳои массив ҳам тавассути ишорагар ҳам тавассути зернавис метавонанд дастраси исти-фодабаранда гарданд. Ҳоло бошад, ин амалро барои векторҳо тавассути итераторҳо анҷом медиҳем. Наҳви муаррифӣ намудани итератор ба намуди зерин аст:

vector <навъи вектор>::iterator номи итератор =қимати аввалаи итератор

Мисол. Истифодаи итераторҳо барои дастрасӣ ба элементҳои вектор.

```
#include <iostream>
#include <algorithm.h>
#define n 20
int main(int argc, char* argv[])
{
    vector <int> v;
    int a[n];
    int i;
    for (i=0;i<n;i++)
        a[i]=random(25)+1;
    for (i=0;i<n;i++)
        v.push_back(a[i]) ;
    cout<<"Elementhoi vector bo index" <<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    cout<<"Elementhoi vector bo iterator:" <<endl;
    vector <int> ::iterator vi=v.begin() ;
    while(vi!=v.end() ){
        cout<<*vi<<" " ;
        vi++;
    }
    return 0;
}
```

Ин мисол ҳамон мисоли қабли буда, танҳо дар ин ҷо итераторе ба намуди зерин муаррифӣ шуда аст.

```
vector <int> ::iterator vi=v.begin() ;
```

Ба сифати қимати аввала итератор ба аввали вектор ишора мекунад. Баъд дар дохили ҳалқаи такрорӣ то ба охири вектор расидан давр такрор гардида, ҳар дафъа як элементро ба ҷоп медиҳад. Вақте ки охири вектор намоён гашт, барнома кори худро ба анҷом мерасонад. Дар ин ҷо begin () ва end () ишора ба аввал ва охири вектор мекунад. Натиҷаи барнома намуди зеринро дорад:

Elementhoi vector bo index:

25 5 12 20 13 9 7 10 12 12 12 20 1 8 19 5 14 24 7 21

Elementhoi vector bo iterator:

25 5 12 20 13 9 7 10 12 12 12 20 1 8 19 5 14 24 7 21

Функсияҳои insert() ва erase ()

Тавре ки дар мисоли пешина мушоҳида кардем, функсияи push_back() барои илова намудани элемент фақат ба охири вектор истифода мешуд. Барои дар дилхоҳ ҷойи вектор гузоштани элемент функсияи insert() истифода бурда мешавад ва барои нест кардани элементҳо бошад, аз функсияи erase() истифода бурда мешавад. Ин функсияҳо бо итераторҳо амал мекунанд. Ба барномаи зерин диққат диҳед:

```
#include <iostream>
#include <algorithm.h>
using namespace std;
#define n 20
int main(int argc, char* argv[]){
    vector <int> v;
    int i;
    for (i=1;i<=n;i++)
        v.push_back(i) ;
    cout<<"Andozai vector="<<v.size() <<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" ";
    cout<<endl;
    vector <int>::iterator vi=v.begin() ;
    vi+=10;
    for (i=4;i<=n ; i+=4)
        v.insert(vi,1,i) ;
    cout<<"Andozai navi vector="<<v.size() <<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" ";
    cout<<endl;
    return 0;
}
```

Дар ин мисол, низ аввал ба vector ададҳои фосилаи аз 1 то 20 бахшида шуда, сипас бо номи vi итераторе муаррифӣ гардидааст. Ба сифати қимати аввала ба итератори эълоншуда ибтидои вектор бахшида шудааст.

Дастури vi+=10, онро мефаҳмонад, ки шурӯъ аз элементи даҳуми вектор элемент илова карда шавад.

Дастури

```
for (i=4;i<=n ; i+=4)
    v.insert(vi,1,i)
```

ба вектор аз мавқеи 10-ум ҳамаи ададҳои ба чор каратии фосилаи аз 1 то 20-ро илова мекунад. Адади 1 миқдори такроршавии ҳар як элементро мефаҳмонад. Натиҷаи барнома чунин мебошад:

```
Andozai vector=20
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

```
Andozai navi vector=25
```

```
1 2 3 4 5 6 7 8 9 10 20 16 12 8 4 11 12 13 14 15 16 17 18 19 20
```

Функсияи `erase` бошад, барои аз вектор нест намудани элементҳои нишондодашуда истифода бурда мешавад. Ин функсия низ бо итераторҳо кор мекунад.

Мисол. Истифодаи функсияи `erase` якҷоя бо итераторҳо.

```
#include <iostream>
#include <algorithm.h>
using namespace std;
#define n 20
int main(int argc, char* argv[]){
    vector <int> v;
    int i;
    for (i=1;i<=n;i++)
        v.push_back(i) ;
    cout<<"Andozai vector="<<v.size() <<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    vector <int>::iterator vi=v.begin() ;
    vi+=5;
    v.erase(vi, vi+5) ;
    cout<<"Andozai navi vector="<<v.size() <<endl;
    for (i=0;i<v.size() ;i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    return 0;
}
```

Ин мисол бошад, аз вектори додашуда элементҳои фосилаи аз 5 то 10-ро нест мекунад. Дастури `vector <int>::iterator vi=v.begin()` итераторро бо номи вектор муаррифӣ намуда, ба сифати қимати аввала ба он ибтидои векторро нишон медиҳад. Баъдан дастури `vi+=5` қимати итераторро баробар ба элементи панҷум қарор медиҳад. Сатри `v.erase(vi, vi+5)` бошад, аз вектори нишондодашуда, 5-то элементро сар карда, аз мавқеи аз 5+1-ум нест мекунад. Натиҷаи барнома намуди зеринро дорад:

```
Andozai vector=20
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

```
Andozai navi vector=15
```

```
1 2 3 4 5 11 12 13 14 15 16 17 18 19 20
```

Рӯйхатҳо

Дар классҳои векторҳо, тавре ки мушоҳида кардем, ба элементҳои вектор танҳо аз як тараф метавонем, дастрасӣ дошта бо-шем. Ҳоло бошад, `class`-ро дида мебароем, ки ба элементҳои он мета-вонем ҳам аз пеш ба охир ва ҳам аз

охир ба аввал дастрасӣ дошта бо-шем. Ин class-ҳо класси рӯйхатҳои дусамта ном доранд. Наҳви шаблони умумии ин class-ҳо намуди зеринро дорад:

```
Template <class T, class Allnameor=allnameor <T>> class list>
```

Дар ин ҷо T наҳви додашудае мебошад, ки class-ҳо бо он тип дар ҳофиза сабт карда мешаванд. Allnameor ҳофизаест, ки барои сабт шудани листҳо ҷудо карда мешавад.

Дар класси лист структурҳои зерин истифода бурда мешаванд:

1. explicit list (const allnameor &a=Allnameor());
2. explicit list (size_type num, const T & val=T(), const allnameor &a=allnameor());
3. list (const list<T, allnameor> ab)
4. Template <class InIter> list (InIter start, InIter end, const allnameor &a=allnameor());

Дар қолаби якум наҳви сохтани рӯйхати ҳолӣ нишон дода шудааст. Форма ё қолаби дуюм барои сохтани рӯйхати дорои num элемент, ки ҳар элементи он ба қимати val баробар мебошанд, истифода мешаванд. Қолаби сеюм барои сохтани рӯйхате, ки элементҳои он аз объекти ab гирифта шудаанд, пешбинӣ шудааст. Қолаби охири бештар барои сохтани рӯйхате, ки элементҳои он дар порчаи аз start то end мебошанд, истифода мешаванд. Монанди класси vector класси лист низ дорои функсия-аъзоҳои зиёде мебошад, ки баъзе аз онҳо дар ҷадвали зерин гирд оварда шудаанд. Функсия-аъзоҳои аз ҷама зиёдтар истифода мешаванд: функсия-аъзои push_front()-барои гузоштани элементҳо ба аввали class, push_back()- барои гузоштани аъзоҳо ба охири class, merge –барои якҷоя кардани ду рӯйхат ва ғайра.

№	Функсия (Функция-член)	Маъно ва мафҳум
1	template <class InIter> void assign (InIter start, InIter end)	Пайдарпайие, ки дар фосилаи аз start то end муайяншударо дар вектор мегузорад.
2	void assign (size_type num, const T &val)	Дар list num-то элемент бо қимати дар параметри val нишондодашуда мегузорад
4	reference back(); const_reference back() const;	Ба элементи охири list нишонагар бармегардонад.
5	iterator begin(); const_iterator begin() const;	Барои элементи яки list итератор бармегардонад.
7	void clear();	Ҳамаи элементҳои list -ро нест мекунад.
8	bool empty() const;	Агар list элемент надошта бошад, қимати ҳақ бармегардонад.
9	iterator end(); const_iterator end() const;	Барои элементи охири list итератор бармегардонад.
10	iterator erase (Iterator i)	Элементро, ки дар параметри i суроғаш нишон дода шудааст, аз рӯйхат нест намуда, ба элементи пас аз он итератор бармегардонад.
11	iterator erase (Iterator start,	Элементҳое, ки дар фосилаи аз start то

	Iterator end);	end нишондода шударо аз list нест мекунад.
12	reference front(); const reference front() const;	Барои элементи якум дар list нишонагар бармегардонад
13	allnameor_type get allnameor () const;	Ҳофизаи барои list ҷудошударо бармегардонад.
14	iterator insert (Iterator I, const T & val=T());	Пеш аз элементе, ки дар параметри I нишон дода шудааст қимати val-ро гузошта ба он итератор бармегардонад.
15	void insert (Iterator I, size_type num, const T &val)	Пеш аз элементе, ки дар параметри val нишон дода шудааст, num нусха қимати val –ро мегузорад
16	Template <class InIter> void insert (Iterator I, InIter start, InIter end);	Пеш аз элементе, ки дар параметри I нишон дода шудааст, пайдарпайии элементҳои аз start to end –ро мегузорад.
17	size_type max_size() const	Барои муайян намудани элементи максималии list хизмат мекунад .
18	void merge (list <T, Allnameor> &ob); Template <class comp> void merge (List <T,Allnameor> &ob, comp cmpfn);	List-ҳои мурағтабро дар намуди мурағтаб якҷоя мекунад. Дар қолаби дуҷум бошад, бояд нишон дода шавад, ки аз рӯи кадом қоида бояд листҳо мурағтаб карда шаванд.
19	void pop_ back()	Элементи охири list-ро нест мекунад.
20	void pop_ front()	Элементҳои додашударо дар аввали лист менависад.
21	void push_ _back(const T &val)	Элементе, ки дар параметри val нишон дода шудааст, дар охири list мегузорад.
22	void push_ front(const T val)	Элементе, ки дар параметри val нишон дода шудааст, дар аввали list мегузорад.
23	reverse Iterator rbegin(); const rereverse_ iterator rbegin () const	Итератори реверсивиро ба охири list бармегардонад.
24	reverse Iterator rend(); const rereverse_ iterator rend () const	Итератори реверсиви ба аввали list бармегардонад.
25	void remove (const T,&val)	Аз list элементеро ба қимати val нест мекунад.
26	Template <class unPred> void remove_ if(unPred pr);	Элементеро, ки дар он передикати унари дорой қимати true аст, аз list нест мекунад.
27	void resize (size_type num, T val=T());	Андозаи list-ро баробар ба қимати num қарор медиҳад. Агар list барои зарурати васеъ шудан эҳтиёҷ дошта бошад, пас,

		дар охири он элементҳои ба қимати val баробар илова карда мешавад.
28	void resize (size_type num T val=T())	Андозаи list-ро num мекунад. Агар list аз num камтар элемент дошта бошад, пас, элементе, ки дар параметри val нишон дода шудааст, дар охири list ҷойгир мекунад.
29	size_type size() const;	Миқдори ҷорӣ list-ро бармегардонад.
30	void start (); template <class comp> void sort(comp cmpfn)	Фармони якум барои мурағтаб намудани элементҳои list истифода бурда мешавад, фармони дуввум бошад, рӯйхатеро тавассути функсияи cmpfn мурағтаб мекунад. Дар ин сурат бояд машаххас карда шавад, ки элементҳо аз рӯи кадом қонуният бояд мурағтаб гарданд.
32	void splice(iterator , list <t, allnameor > &ob);	Дар маҷкеи i-уми list-и даъватшаванда объекти ob-ро мегузорад. Баъди иҷрои ин амал list холи мегардад.
33	void splice (iterator i , list <t, allnameor > &ob, iterator el)	Аз list элементҳоеро, ки суроғаи онҳо дар параметри el нишон дода шудааст, нест намуда, дар list-и даъватшаванда дар маҷкее, ки суроғаи он дар параметри I нишон дода шудааст, менависад.
34	void splice(Iterator I, list <T, allnameor> &ob, Iterator start, Iterator end)	Аз list элементҳоеро, ки дар параметри аз start to end нишон дода шудаанд, нест намуда, ба рӯйхати даъватшаванда сар карда, аз маҷкеи i-юм менависад.
36	void swap (deque<T,Allnameor> &ob);	Элементҳои вектори даъватшударо ба элементҳои ob иваз мекунад.
37	void unique (); Template <class BinPred> void unique (BinPred pr)	Фармони аввал аз list элементҳоеро, ки ду маротиба омадаанд, нест мекунад. Фармони дуввум бошад, барои мушаххас намудани он, ки предикати pr ягона аст, хизмат мекунад.

Мисол. Истифодаи рӯйхатҳо.

```
#include <iostream>
#include <algorithm.h>
#include <list.h>
using namespace std;
int main(int argc, char* argv[]){
    list <int> l;
    int i;
    for (i=1; i<=20;i+=2)
        l.push_back(i);
    cout<<"Andozai list="<<l.size()<<endl;
```



```

    cout<<"Elementhoi list:"<<endl ;
    list <int>::iterator p;
    p=l.begin() ;
    while (p!=l.end() ){
        cout<<*p<<" ";
        p++;
    }
    return 0;
}

```

Тавре ки дар ин ҷо дида мешавад, дар аввал List-е бо номи l муаррифӣ шудааст, ки дорои эгон элемент намебошад. Баъд дар дохили ҳалқа, тавассути функсияи push_bask() ба List элемент бах-шида шудааст. Пас аз иҷрои ин амал итераторе муаррифӣ гардида, элементҳои list-ро ба ҷоп медиҳад. Ба қитъаи барномаи зерин нигаред:

```

list <int>::iterator p;
p=l.begin() ;
while (p!=l.end() ){
    cout<<*p<<" ";
    p++;
}

```

Дар ин қисмати барнома, итераторе ба номи p муаррифӣ карда шуда, ба сифати қимати аввала ба он элементҳои List бахшида шудааст. Функсияи begin() ибтидои List-ро нишон медиҳад ва функсияи end() бошад, ба интиҳои List ишорат мекунад. Аз ин рӯ, дар дохили ҳалқа то ба интиҳои List расидан, ҳар маротиба як элементҳои List ба ҷоп бароварда мешавад. Натиҷаи барнома намуди зеринро дорад:

```

Andozai list=10
Elementhoi list:
1 3 5 7 9 11 13 15 17 19

```

Тавре ки ишора карда будем, ба элементҳои List метавонем аз ду тараф дастрасӣ дошта бошем, ҳам аз аввал ба охир ва ҳам аз охир ба аввал. Инчунин List ҳоро List ҳои дусамта меноманд. **Мисол:**

```

#include <iostream>
#include <algorithm.h>
#include <list.h>
using namespace std;
int main(int argc, char* argv[]){
    list <int> l;
    list <int> l1;
    int i;
    for (i=1; i<=20;i+=2)
        l.push_back(i);
    list<int>:: iterator p;
    cout<<"Elementhoi List az chap ba rost.\n";
    while (!l.empty() ){
        p=l.begin() ;
        cout<<*p<<" ";
        l1.push_front(*p);
    }
}

```

```

        l.pop_front() ;
    }
    cout<<"Elemenyhoi List az rost ba chap.\n";
    p=l.begin() ;
    while (p!=l.end() ){
        cout<<*p<<" ";
        p++;
    }
    return 0;
}

```

Натиҷаи барнома намуди зеринро дорад:

```

Elemenyhoi List az chap ba rost.
1 3 5 7 9 11 13 15 17 19
Elemenyhoi List az rost ba chap
19 17 15 13 11 9 7 5 3 1

```

Мураттабкунии руйхатҳо

Элементҳои List-ҳо тавассути функсияи sort мураттаб карда мешаванд. Дар мисоли зерин тарзи истифодаи ин функсия оварда шудааст. Мисол:

```

#include <iostream>
#include <algorithm.h>
#include <list.h>
int main(int argc, char* argv[]){
    list <int> l;
    int i;
    for (i=1; i<=20;i++)
        l.push_back(random(50)+1);
    cout<<"Andozai list="<<l.size()<<endl<<endl;
    cout<<"Elementhoi list pesh az murattabkuni:"<<endl;
    list <int>::iterator p;
    p=l.begin() ;
    while (p!=l.end() ){
        cout<<*p<<" ";
        p++;
    }
    cout<<endl<<endl;
    cout<<"Elementhoi list pas az murattabkuni:"<<endl;
    l.sort() ;
    p=l.begin() ;
    while (p!=l.end() ){
        cout<<*p<<" ";
        p++;
    }
    return 0;
}

```

Ин мисол низ монанди мисоли 2 буда, фарқиаш дар он аст, ки дар ин мисол элементҳои List ба таври тасодуфӣ интихоб карда шудаанд ва тавассути функсияи sort() мураттаб карда шудаанд. Дар ин мисол, аввал элементҳои List

бароварда шудаанд, ки пас аз даъвати функцияи sort() бори дигар ба чоп дода шудаанд. Пас аз даъвати функцияи sort элементҳои List дар ҳама ҷои барнома, пас аз мавқеи даъватшавии функция мураттаб мебошанд. Натиҷаи барнома чунин аст:

```
Andozai list=20
Elementhoi list pesh az murattabkuni:
50 5 37 45 13 34 7 35 37 12 12 45 1 33 44 30 14 49 7 46
Elementhoi list pas az murattabkuni:
1 5 7 7 12 12 13 14 30 33 34 35 37 37 44 45 45 46 49 50
```

List-ҳо аз массив

Ба сифати элемент ба List-ҳо метавонанд, массивҳо низ иштирок кунанд. Ҳоло исботи ин гуфтаро дар мисоли зерин дида мебароем Мисол:

```
#include <iostream>
#include <algorithm.h>
#include <list.h>
using namespace std;
#define n 35
int main(int argc, char* argv[]){
    int arr[n];
    list <int> l;
    int i;
    for (i=0; i<n;i++)
        arr[i]=(rand()%50);
    for (i=0; i<n;i++)
        l.push_back(random(arr[i]));
    cout<<"Andozai list="<<l.size()<<endl;
    cout<<"Elementhoi list :"<<endl ;
    list <int>::iterator p;
    p=l.begin() ;
    while (p!=l.end() ){
        cout<<*p<<" ";
        p++;
    }
    return 0;
}
```

Тавре ки мебинем, дар ин ҷо массиве ба номи arr муаррифӣ гардида, ба ададҳои тасодуфӣ пур карда шудааст. Дар дохили ҳалқаи такрорӣ элементҳои массив ба List бахшида шудаанд. Натиҷаи барнома намуди зеринро дорад:

```
Andozai list=20
Elementhoi list :
25 5 21 5 15 8 7 29 9 0 2 6 21 14 4 1 1 1 29 10
```

Якҷоя кардани ду List

Метавонем ду List-ро, ки аз як навъ мебошанд, якҷоя кунем, барои ин функцияи merge истифода бурда мешавад. Пас аз даъвати ин функция

элементҳои ду List-и якҷо карда шуда, мурағтаб намоиш дода мешаванд.

Мисол:

```
#include <iostream>
#include <algorithm.h>
#include <list.h>
using namespace std;
#define n 35
int main(int argc, char* argv[]){
    int arr[n];
    list <int> l;
    list <int> l1;
    int i;
    for (i=0; i<n;i++)
        l.push_back(2*i);
    for (i=0; i<n;i++)
        l1.push_back(2*i+1);
    cout<<"Andozai list="<<l.size()<<endl<<endl;
    cout<<"Elementhoi list :"<<endl<<endl ;
    list <int>::iterator p;
    p=l.begin() ;
    while (p!=l.end() )
    {
        cout<<*p<<" ";
        p++;
    }
    cout<<endl;
    p=l1.begin() ;
    while (p!=l1.end() )
    {
        cout<<*p<<" ";
        p++;
    }
    cout<<endl<<endl;
    l.merge(l1);
    if (l1.empty() )
        cout<<"list holi shud!"<<endl;
    p=l.begin() ;
    while (p!=l.end() ){
        cout<<*p<<" ";
        p++;
    }
    return 0;
}
```

Дар ин мисол тавре ки мушоҳида мегардад, ду List муаррифӣ гардидаанд. Барои List-и якум ба сифати элемент, ададҳои ҷуфт ва ба List-и дуввум, ададҳои тоқӣ фосилаи аз 0 то 35 бахшида шудаанд. Дар аввал элементҳои ин List-ҳо ба ҷоп бароварда мешаванд. Сипас, функсияи merge даъват карда мешавад. Функсияи мазкур List-и дуввумро дар листи якум нусхабардорӣ мекунад. Ҳар гоҳ, ки List-и дуввум аз элемент холи шавад, пайғоми муносибе ҷоп карда

мешавад. Натиҷаи барнома List-и дуввум дар якум нусхабардорӣ карда мешавад. Дар ҳангоми нусхабардорӣ инчунин мурағаб низ карда мешавад. Натиҷаи барнома намуди зеринро дорад:

```
Andozai list=20
Elementhoi list :
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39
list holi shud!
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
26 27 28 29 30 31 32 33 34 35 36 37 38 39
```

Тасвирҳо (Тачассумҳо)

Дар библиотекаи map иттиҳоди контейнерҳо мавҷуданд, ки ҳар кадоми онҳо дорои калид ва қимат мебошанд. Калид ин номи муқаррарии элемент буда, тавассути он метавон ба қимати он дастрас шуд. Тасвирҳо монанд ба фарҳанги луғат мебошанд, яъне бо дониستاني калима метавонем маънои онро низ ба даст оварем. Мисоли дигар калид ин номи одам ва қимат маълумотҳои ҳамон одам мебошанд, ки дониستاني номи муштари метавонем аз пойгоҳи додаҳои телефон дигар маълумотҳои муштариро ба даст оварем. Дар барноманависӣ тасвирҳо нисбатан маъмултар буда, истифодаи зиёдтар доранд. Калидҳо бояд, ки ягона бошанд, дар сурати такрор шудани калид ба кори барнома халал ворид мешавад. Аз ин рӯ, роҳи дигарро пеш мегирем, барои калидҳои, ки такроран меоянд, аз қолаби multimap истифода мебарем.

Тарзи истифодаи қолаби map шакли зеринро дорад:

```
Template <class key, class T,
      Class comp=Less <key>,
      Class allnameor =allnameor<pair < const key, T>>> class
map
```

Дар ин ҷо key –навъи додаҳои калид, T навъero ифода мекунад, ки тасвирҳо бо он ном сабт мешаванд. Comp –функсияе мебошад, ки ду калидро муқоиса мекунад. Ба таври пешфарз ба ҷои параметри comp функсияи Less истифода бурда мешавад. Параметри allnameor бошад ҳофизаero барои тасвирҳо ҷудо ё мушаххас мекунад.

Монанди дигар class-ҳо классии map низ дорои конструкторҳои зерин мебошад.

1. Explicit map (const comp &compfn=comp(), Const allnameor &a=allnameor());
2. map (const map < key , T, comp, Allnameor> &ob)
3. Template <class InIter> map (InIter start, InIter end, const Comp &compfn=Comp(), const Allnameor & a=Allnameor());

Формаи аввал конструктор тасвири холиero эҷод мекунад. Қо-лаби дуввум барои сохтани тасвире, ки элементҳои он аз Ob гирифта шудаанд, истифода бурда мешавад. Қолаби сеюм бошад, барои эҷод намудани тасвире, ки қиматҳои он дар фосилаи аз start to end мехобанд пешбинӣ шудааст. Параметри

string барои муайян намудани ҳаракристикаи мурағтабукунии элементҳои тасвирҳои истифода бурда мешавад.

Ҳамаи объектҳои, ки барои онҳо Операторҳои муқоисакунии истифода бурда мешаванд, метавонанд, ба сифати калид хизмат кунанд. Амалҳои муқоисакунии намуди зеринро доранд:

==, <=, >=, !=, >, < ,

Қолаби классии калид ва қимат “pair” намуди зеринро дорад:

```
Template <class ktype , class vtype > struct pair {
    Typedef ktype first_type; // Навъи калид
    Typedef vtype second_type ; // Навъи қимат
    ktype first ;
    vtype second;
    //Конструктор
    pair();
    pair (const ktype &k, const vtype &v);
    Template <class A, class B> pair (const <A, B>
    &ob);
}
```

Дар ин қолаб калимаи first нишондиҳандаи калид ва калимаи second бошад, нишондиҳандаи қимат мебошад. Ва чи хеле, ки айён аст, ҷуфти калид ва қимат дар дохили як сохтор иҷро мегардад.

Илова бар ин, ҷуфти калид ва қиматро метавон, тавассути функцияи make_pair() дар асоси навъи додаҳо сохт. Функцияи мазкур хулоса ё ҷамъбасти функцияҳо мебошад. Ҷуфти калид ва қимат дар қолаби функцияи make_pair() намуди зеринро дорад:

```
Template <class ktype , class vtype >
pair <ktype, vtype> make_pair(const ktype &k, const vtype &v) ;
```

Мисоли 1. Барномае нависед, ки ба сифати калид ҳарфҳо ва ба сифати қимат ASCII коди ҳарфҳо дар худ нигоҳ дорад. Бо дохил кардани рамз коди онро нишон диҳад.

```
#include <iostream>
#include <algorithm.h>
#include <list.h>
#include <map.h>
#define n 26
using namespace std;
int main(int argc, char* argv[]){
    int arr[n];
    map <char, int> m;
    int i;
    for (i=32; i<n;i++)
        m.insert(pair<char, int>(i, i));
    char ch;
    cout<<"Kalidro vorid kuned"<<endl<<endl;
    cin>>ch;
    map<char,int>::iterator p;
```

```

p=m.find(ch);
if (p!=m.end() )
    cout<<p->second ;
else
    cout<<"Chunin kalid mavjud nest!!!!!!!!!"<<endl;
return 0;
}

```

Дар ин мисол дар аввал барои дастрас шудан ба ҷуфти калид ва қимат, библиотекаи <map> пайваст карда шудааст. Баъдан дар барнома ҷуфти калид ва қимат, ки як аргументи он рамзӣ ва дигараш бутун мебошад, ба намуди

```
map <char, int> m
```

муаррифӣ карда шудааст. Дар дохили ҳалқаи такрори ба ин ҷуфт рамзҳои ASC код ва қиматҳои он тавассути функсияи insert ба тасвир гузошта шудаанд. Сипас итераторе ба номи p муаррифӣ шудааст. Ин тасвир тавассути функсияи find() то замони ба охири тасвир расидан калиде, ки ворид карда шудааст, ҷустуҷӯ мегардад. Агар калид пайдо шуд, қимати онро дар саҳифаи тасвир ба чоп мефиҳад ва агар калид мавҷуд набошад, пайгоми муносибе ба чоп мефиҳад. Барои дохил кардани калид тағйирёбандае аз навъи char муаррифӣ карда шудааст.

Монанди дигар class –ҳо класси map низ дорои функсияҳои зиёде мебошад, ки дар ҷадвали зерин гирд оварда шудаанд.

№	Функсия (Функция -член)	Маъно ва мафҳум
1	iterator begin(); const_iterator begin() const;	Барои элементи якуми тасвир итератор бармегардонад.
2	void clear();	Ҳамаи элементҳои тасвирро нест мекунад.
3	bool empty() const;	Агар тасвир элемент надошта бошад қимати ҳақ бармегардонад.
4	iterator end(); const_iterator end() const;	Барои элементи охири тасвир итератор бармегардонад.
5	void erase (Iterator i)	Элементро, ки дар параметри I суроғаш нишон дода шудааст аз рӯйхат нест намуда, ба элементи пас аз он итератор бармегардонад.
6	void erase (Iterator start, iterator end);	Элементҳое, ки дар фосилаи аз start то end нишон дода шудаанд, аз тасвир нест мекунад.
7	size_type count(const key_type &k) const)	Рақами калиди воридшударо ба-рои тасвир бармегардонад (1 ё 0)
8	size_type erase (const key_type &k)	Элементи тасвирро, ки калиди он ба қимати k доро аст, аз тасвир нест мекунад.
9	iterator find(const key_type &k) const_iterator find(const key_type &k) const	Итератор қимати калиди додашуда-ро бармегардонад. Агар қимати калид мавҷуд набошад, охири тасвирро

		бармегардонад.
10	<code>allnameor_type get_allnameor() const</code>	Ҳофизаи барои тасвир ҷудокардашударо бармегардонад.
11	<code>iterator insert (iterator i, const value_type &val)</code>	Дар мавқеа, ки итератори I нишон дода шудааст, қимати val-ро мегузорад ва ба ин элемент итератор бармегардонад.
12	<code>template <class initer> void insert(initer start, initer end)</code>	Элементҳои додашударо дар фосилаи [start; end] мегузорад.
13	<code>pair <iterator, bool> insert(const value_type & val)</code>	Дар тасвири даъватшуда қимати val-ро мегузорад ва ба он итератор бармегардонад. Дар ҳамон вақт метавонем элементе гузошт, ки агар дар тасвир вуҷуд надошта бошад. Дар ҳолати вуҷуд доштани элемент қимати <code>pair <iterator, true></code> ва дар ҳолати вуҷуд надоштан қимати <code>pair< iterator, false></code> бар-гардонида мешавад.
14	<code>key compare key_comp() const</code>	Объект_функсия қиматеро-ро бармегардонад, ки ба калид мувофиқат кунад.
15	<code>iterator lower_bound(const key_type &k); const_iterator lower_bound (const key_type &k) const;</code>	Барои элементҳои якуми тасвир, ки калид барои он ба қимати k баробар аст ё аз он элемент калонтар аст бармегардонад.
16	<code>size_type max_size const();</code>	Қимати максималии элементҳои, ки дар тасвир мавҷуд ҳастанд бармегардонад.
17	<code>reference operator [] (const key_type &i)</code>	Нишонагар ба элементе, ки дар параметри I нишон дода шудааст бармегардонад. Агар чунин элемент вуҷуд надошта бошад дар тасвир онро мегузорад.
18	<code>reverse_iterator rend(); const_reverse_iterator rend() const</code>	Ба ибтидои тасвир итератори реверсивӣ бармегардонад.
19	<code>reverse_iterator rbegin(); const_reverse_iterator rbegin() const</code>	Ба интиҳои тасвир итератори реверсивӣ бармегардонад.
20	<code>size_type size() const;</code>	Миқдори элементҳои ҷорӣ тасвирро бармегардонад.
21	<code>void swap(map_key, T, comp, Allnameor> &ob)</code>	Барои объекти даъватшавандаро ба объекти дар параметри ob иваз намудани хизмат мекунад.
22	<code>iterator upper_bound(const key_type &k);</code>	Барои элементҳои тасвир итератор бармегардонад, агар қимати калид аз

	const_iterator upper_bound(const key_type &k) const;	қимати додашудаи к калон бо-шад.
23	value_compare value_comp() const;	Объект_функсия қимати муқоиса-кардашударо бармегардонад.

Мисоли 2. Барномае нависед, ки амали ҷустуҷӯи калимаро дар дохили фарҳанг (словар) ҷустуҷӯ кунад.

```
#include <iostream>
#include <map.h>
using namespace std;
#define n 20
typedef map <string, string> mymap;
typedef mymap::const_iterator iter;
int main(int argc, char* argv[])
{
    mymap dict;
    dict["father"]=" padar";
    dict["mather"] = "modar";
    dict["brather"] = "barodar";
    dict["sister"] = "khohar";
    dict["grandmather"] = "modarkalon";
    string word; // калима барои ҷустуҷӯ дар дохили map
    cout << "Enter word: ";
    cin >> word;
    iter i = dict.find(word); // Ҷустуҷӯ
    if(i != dict.end()) // агар калима ёфт шавад
        cout << "Manoi kalimai"<<word << " " << i->second<<"
meboshad." << endl;
    else
        cout << "word not found!" << endl;
    return 0;
}
```

Дар инҷо аввал typedef-и класи map ва итератор муаррифӣ гардидааст, сипас дар методи асосӣ бо номи dict тасвири муаррифӣ гардидааст. Баъд аз ин ба тасвири муаррификардашуда қимат бахшида шудааст. Барои ҷустуҷӯи калимаи воридкардашуда аз функсия-аъзои find истифода карда шудааст.

Мисоли 3. Хориҷ кардани элементҳои таҷассум.

```
#include <iostream>
#include <map.h>
using namespace std;
typedef map<string, string> mymap;
typedef mymap::const_iterator iter;
int main(int argc, char* argv[])
{
    mymap dict;
    dict["father"]=" padar";
    dict["mather"] = "modar";
    dict["brather"] = "barodar";
    dict["sister"] = "khohar";
```

```

dict["grandmather"] = "modarkalon";
cout<<"Dictionar"<<endl;
iter i ;
for(i=dict.begin();i!=dict.end();i++)
    cout<<i->first<<":"<<i->second<<endl;
return 0;
}

```

Барои хориҷ кардани элементҳои таҷассум аз итераторҳо истифода бурда шудааст.

Мисоли 4. Нест кардани элементҳои таҷассум.

```

#include <iostream>
#include <algorithm.h>
#include <map.h>
using namespace std;
typedef map<string, string> mymap;
typedef mymap::const_iterator iter;
int main(int argc, char* argv[]){
    mymap dict;
    dict["father"]=" padar";
    dict["mather"] = "modar";
    dict["brather"] = "barodar";
    dict["sister"] = "khohar";
    dict["grandmather"] = "modarkalon";
    cout<<"Dictionar"<<endl;
    iter i ;
    for(i=dict.begin();i!=dict.end();i++)
        cout<<i->first<<":"<<i->second<<endl;
    string word; // калима барои нест кардани элементҳои map
    cout << "Enter word for deleted ";
    cin>>word;
    if(dict.find(word) != dict.end())
        dict.erase(word);
    for(i=dict.begin();i!=dict.end();i++)
        cout<<i->first<<":"<<i->second<<endl;
    return 0;
}

```

Барои нест кардани элементҳои таҷассум аз функсия-аъзои `erase` истифода бурда шудааст.

Алгоритмҳо

Алгоритмҳои зиёди коркарди додаҳо дар контейнерҳо мавҷу-данд. Чи хеле ки медонем, ҳар як контейнер амалҳои махсуси база-виро пуштибонӣ мекунад. Алгоритмҳои стандарти нисбат ба дигар контейнерҳо истифодаи васеътар доранд. Барои ба объектҳои классҳои мазкур дастрас шудан, бояд ки библиотекаи `<algorithm.h>` дар барнома пайваست карда шавад. Дар библиотекаи мазкур функсияҳои зиёде мавҷуданд, ки муҳимтаринашон дар ҷадвали зерин гирд оварда шудаанд.

№	Функсия (Функция-член)	Маъно ва мафҳум
1	objecent_find()	Дар пайдарпайии элементҳо, элементи мавриди назарро ҷустуҷӯ мекунад ва ба элементи аввала итератор бармегардонад.
2	binary_search()	Дар пайдарпайии мураттаб амали ҷустуҷӯро иҷро мекунад.
3	copy()	Пайдарпаiero нусхабардори меку-над.
4	copy_backward()	Монанди функсияи copy буда, фарқияташ дар он аст, ки он элементҳоро аз охир нусхабардорӣ мекунад.
5	count()	Миқдори элементҳои пайдарпаiero майян мекунад.
6	count_if()	Миқдори элементҳое, ки шarti предикатро муайян мекунанд ҳисоб мекунад.
7	equal()	Якхелагии (баробарии) ду фосиларо муайян мекунад.
8	equal_range()	Фосилаero бармегардонад, ки ҳангоми ба он гузоштани элемент низоми мураттабии пайдарпаи халалдор намешавад.
9	fill ва fill_n()	Ба қимати додашуда фосилаи нишондодашударо пур мекунад.
10	find()	Дар фосилаи нишондодашуда қи-мати додашударо ҷустуҷӯ мекунад ва итератор ба элементи ҷустуҷушаванда бармегардонад.
11	find_end()	Монанди find буда танҳо элементи охирони ҷустуҷушавандаро бармегардонад.
12	find_first_of()	Амали ҷустуҷӯро барои элементи якум дар дохили пайдарпаии додашуда иҷро мекунад, ки он ба дилхоҳ элементи фосилаи додашуда мувофиқат мекунад.
13	find_if()	Дар фосилаи додашуда амали ҷустуҷӯиро иҷро намуда, ба элементе, ки истифодабаранда барои он передикати ягона муайян кардааст қимати true бармегардонад.
14	for_each()	Барои фосилаи нишондодашуда ин функсия иҷро мегардад.
15	generate ва generate_n()	Элементҳое, ки функсия итератор аз фосилаи додашуда баргардонидааст, муқоиса мекунад.
16	inplace_merge()	Як фосилаи додашударо ба дигараш якҷоя

		мекунад.
17	iter_swap()	Қои қиматҳоро, ки итератор онҳоро ба сифати аргумент баргардонидааст, иваз мекунад.
18	lexicographical_compare()	Як пайдарпайи додашударо бо дигараш дар тартиботи лексикографӣ муқоиса мекунад.
19	lower_bound()	Амали ҷустуҷӯро барои элементи яқум дар пайдарпайии додашуда, ки он аз элементи додашуда хурд намебошад, амалӣ мегардонад.
20	max()	Аз ду элемент калонтаринашро бармегардонад.
21	max_element()	Барои элементҳои максималӣ дар фосилаи додашуда итератор бармегардонад.
22	merge()	Ду пайдарпайии мурағтабро якҷоя мекунад. Пайдарпайии ҳосилшударо низ мурағтаб мекунад.
23	min()	Аз ду қимати додашуда хурдтаринашро бармегардонад.
24	min_element()	Барои элементҳои минималӣ дар фосилаи додашуда итератор бармегардонад.
25	mismatch()	Амали ҷустуҷӯро барои элементҳои якхела набудаи ду пайдарпайи иҷро намуда, ба он элементҳо итератор бармегардонад.
26	partial_sort()	Элементҳои фосилаи додашударо мурағтаб мекунад.
27	partition()	Фосилаи додашударо аз рӯи шартӣ зерин мурағтаб мекунад. Ҳамаи элементҳое, ки дар предикат дода шудаанд, қимати true бармегардонад ва барои элементҳое, ки пеш аз предикат омадаанд, қимати false бармегардонад.
28	pop_heap()	Қои элементи яқум ва яқто кам аз охири пайдарпайи додашударо иваз намуда, баъд онро барқарор мекунад.
29	remove(), remove_if() , remove_copy() ва remove_copy_if()	Аз фосилаи додашуда элементҳоро нест мекунад.
30	replace(), replace_if(), replace_copy() ва replace_copy_if()	Элементҳои фосилаи додашударо ба дигар элементҳо иваз мекунад.
31	rotate ва rotate_copy()	Дар фосилаи додашуда амали лағжиши даврӣ аз рост амалӣ мекунад.

32	reverse va reverse_copy()	Элементҳои фосилаи додасударо ба бараксаш иваз мекунад.
33	search()	Амали ҷустуҷӯии як пайдарпаиро дар дохили дигараш иҷро мекунад.
34	searche_n()	Дар дохили якчанд пайдарпайи элементи монандро ҷустуҷӯ мекунад.
35	set_difference()	Аз фарқи ду пайдарпайи пайдарпайии нав месозад.
36	set_intersection()	Аз бурриши ду пайдарпаи пайдар-пайии нав месозад.
37	set_symmetric_difference()	Аз фарқи симметрии ду пайдарпайи пайдарпайии нав месозад.
38	set_union()	Аз якҷоякунии ду пайдарпайии мураттаб пайдарпайии нав месозад.
39	sort()	Элементҳои фосилаи додасударо мураттаб мекунад.
40	sort_heap()	Амали мураттабкунии Кӯчишро дар фосилаи додасуда амалӣ мекунад.
41	stople_partition()	Пайдарпайии додасударо аз рӯи шарти зерин мураттаб мекунад: Барои ҳамаи элементҳои, ки дар передикат додасудаанд, қимати true бармегардонад ва барои элементҳои, ки пеш аз предикат омадаанд, қимати false бармегардонад. Чунин мураттабкунӣ мураттабкунии устувор номида мешавад.
42	stable_sort()	Амали мураттабкунии устуворро ба-рои фосилаи додасуда амалӣ мекунад. Ин мураттабкунӣ маънои онро дорад, ки элементҳои якхела ба назар гирифта намешаванд.
43	swap()	Ҷойи ду қимати додасударо иваз мекунад.
44	swap_ranges()	Амали мубодилаи элементҳоро дар фосилаи додасуда иҷро мекунад.
45	transform	Функсияро дар фосилаи додасуда татбиқ намуда, натиҷаҳоро дар пайдарпайии нав гирд меоварад.
46	unique va unique_copy()	Дар фосилаи додасуда элементҳоеро ки такроран омадаанд нест мекунад.
47	upper_bound()	Аз пайдарпайии додасуда элементи охириро, ки аз элементи додасуда калон намебошад ҷустуҷӯ мекунад.

Ҳамаи алгоритмҳои забони C++ ба гурӯҳҳои зерин тақсим карда мешаванд:

1. Алгоритмҳои чустучӯи. Ба ин гурӯҳ шонздаҳо алгоритми гуногун дохил мешавад, ба монанди:

1. max_element()
2. min_element()
3. count()
4. count_if()
5. find()
6. find_if()
7. binary_search()
8. lower_bound()
9. upper_bound()
10. search_n()
11. search()
12. equal_range()
13. find_end()
14. find_first_of()
15. adjacent_find()
16. mismatch()

2. Алгоритмҳои мураттабкунӣ ва ҷойивазкунӣ. Ин гурӯҳ алгоритмҳо имконияти бо усулҳои гуногун мураттаб намудани элементҳои контейнерро фароҳам меоранд. Алгоритмҳои гурӯҳи маз-кур ба элементҳои рӯйхат (List) татбиқ карда намешаванд. Ба ин гурӯҳ алгоритмҳо дохил мешаванд:

1. sort()
2. stable_sort()
3. partial_sort()
4. partial_sort_copy()
5. next_permutation()
6. reverse()
7. reverse_copy()
8. rotate()
9. rotate_copy()
10. partition()
11. stable_partition()
12. random_shuffle()
13. prev_permutation()
14. nth_element()

3. Алгоритмҳои пуркунӣ ва тағйирдиҳӣ. Ин гурӯҳ алгоритмҳо барои сохтан, дигаргун кардан ва ё пур намудани пайдарпаиҳо истифода мешаванд. Ба ин гурӯҳ алгоритмҳои зерин дохил мешаванд:

1. fill()
2. fill_n()
3. generate()
4. generate_n()
5. for_each()
6. transform()

4. Алгоритмҳои несткунӣ ва табдилдиҳӣ. Ин гурӯҳ алгоритмҳо, барои иваз ё нест намудани як ё як фосилаи додашудаи элементҳо истифода бурда мешаванд. Ба ин гурӯҳ алгоритмҳо дохил мешаванд:

1. copy()
2. copy_backwards()
3. remove()
4. remove_copy()
5. remove_if_copy()
6. unique()
7. unique_copy()
8. replace()
9. replace_copy()
10. replace_if()
11. replace_copy_if()
12. iter_swap()
13. swap()
14. swap_range()

5. Алгоритмҳои муқоисакунӣ. Ин гурӯҳ алгоритмҳо асосан ба-рои як контейнерро ба контейнери дигар муқоиса кардан истифода бурда мешаванд. Ба ин гурӯҳ алгоритмҳо дохил мешаванд:

1. max()
2. min()
3. equal()
4. lexicographical_compare()

6. Алгоритмҳои кор бо маҷмӯҳо. Ба ин гурӯҳ алгоритмҳо дохил мешаванд:

1. set_union()
2. set_intersection()
3. set_difference()
4. set_symmetric_difference()
5. includes()

7. Алгоритмҳои ҳисобӣ. Ин гурӯҳ алгоритмҳо барои иҷро намудани амалиёти ҳисобкунӣ дар контейнерҳо истифода бурда мешаванд. Алгоритмҳои мазкур ҳангоми пайваст намудани библио-текаи numeric ба кор мебароянд. Ба ин гурӯҳ алгоритмҳо дохил мешаванд:

1. accumulate()
2. partial_sum()
3. inner_product()

Истифодаи алгоритмҳо дар амалия

Барои ба хонандагон фаҳмотар шудан, мисоли якчандто алгоритмро дар поён меорем, алгоритмҳои боқимондари бошад, ҳавола ба хонандагон месозем. Ҳангоми навиштани қолаби алгоритмҳо, аз калимаҳои кутӯҳшудаи зерин истифода бурда мешавад:

in-итератор барои хондан (inputiterator);

out- итератор барои навиштан (outputiterator);

for- итератори мавзун (мустақим) (forwarditerator);
 bi - итератор дутарафа (bidirectionaliterator);
 ran- итератори дастрасии тасодуфӣ (randomaccessiterator);
 pred-предикати унарӣ (шартӣ) (predicate);
 binpred- предикати бинарӣ ;
 comp-функсияи муқоиакунӣ (compare);
 op-амалиёти унарӣ ;
 binop-амалиёти бинарӣ .

Алгоритми count ва count_if()

Яке аз амалҳои маъмуле, ки барои ҳамаи пайдарпаиҳо иҷро карда мешавад, ин ҳисоб намудани миқдори элементҳо мебошад. Барои иҷрои ин амал, алгоритмҳои count ва count_if истифода бурда мешаванд. Наҳви умумии алгоритмҳои мазкур намуди зеринро доранд:

```

Template <class InIter, class T> Ptrdiff_t count (InIter start, InIter end, const
T &val);
Template <class InIter, class UnPred> Ptrdiff_t count_if (InIter start, InIter
end, UnPred pfn);
  
```

Алгоритми count миқдори элементҳоро баробари val-ро аз фосилаи додашудаи пайдарпаёӣ бармегардонад. Алгоритми count_if бошад, дар фосилаи додашуда миқдори элементҳоеро, ки дар онҳо предикати ягонаи pfn қимати true дорад, бармегардонад. Навъи Ptrdiff_t монанди навъи бутун, муайян карда мешавад. **Мисол:**

```

#include <iostream>
#include <algorithm.h>
#include <map.h>
#include <ctype.h>
#include <vector.h>
using namespace std;

bool sadonok (char ch){
    ch=tolower(ch);
    if (ch=='a' || ch=='e' || ch=='r' || ch=='i' || ch=='o' ||
ch=='u' || ch=='y')
        return false;
}

int main(int argc, char* argv[]){
    char str[]="Classi algorithm behtarin class meboshad!!";
    int i;
    vector <char> v;
    cout<<"str=";
    for (i=0;str[i];i++)    v.push_back(str[i]);
    for (i=0;i<v.size(); i++)
        cout<<v[i];
    cout<<endl;
    int n;
    n=count(v.begin() , v.end(),'a' );
  
```



```

cout<<"Miqdori harfi a="<< n<< " to "<<endl;
n=count_if(v.begin() , v.end(), sadonok);
cout<<"Miqdori harfhoi gairihamsado="<<n<< " to"<<endl;
return 0;
}

```

Дар ин мисол тавре ки айён аст, дар аввал библиотекаҳои зарурӣ насб карда шудаанд. Баъдан функсияи аз навъи мантиқӣ муаррифӣ шудааст, ки он функсия садонокхоро дар ҷумла муайян менамояд, яъне он функсия қимати false бармегардонад, агар дар ҷумлаи додашуда садонок мавҷуд бошад. Дар функсияи асосӣ бошад, тағйирёбандае аз навъи рамзӣ муаррифӣ шуда, ба он ба сифати қимати аввала, ҷумлаи « Classi algorithm behtar class meboshad!!» бахшида шудааст. Сипас, векторе муарифӣ шуда, ба он ба сифати қимат ин сатр бахшида шудааст. Тағйирёбандаи n, ки аз навъи бутун аст, аввал дар даъвати функсияи count барои муайянкунии миқдори ҳарфи «a» истифода шудааст. Дар даъвати функсияи count_if() бошад, барои дар ҷумлаи додашуда муайян намудани миқдори ҳарфҳои ғайриҳамсадо истифода бурда шудааст. Функсияи count_if() чойи холигиро низ ҳамчун як рамз ҳисоб мекунад. Натиҷаи барнома намуди зеринро дорад:

```

str=Classi algorithm behtar class meboshad!!
Miqdori harfi a=5 to
Miqdori harfhoi gairihamsado=28 to

```

Алгоритми count барои ҳисоб намудани миқдори қимати додашуда, дар дохили пайдарпайи истифода шуда, чунин қолаб дорад:

```

template <class In class T>
typename iterator_traits<In>:: difference_type
count (In first, In last, const T &value)

```

ин алгоритм миқдори қимати value-ро дар дохили пайдарпайи ҳисоб мекунад.

Мисол. Ҳисоб намудани миқдори адади 20 дар дохили пайдарпайи бо истифода аз алгоритми count .

```

#include <iostream>
#include <algorithm.h>
#include <map.h>
#include <vector.h>
using namespace std;
int main(int argc, char* argv[]){
    int x[10]={5,20,33,33,20,23,20,17,18, 3};
    vector <int> v;
    for (int i=0; i<10; i++) {
        cout<<x[i]<<" ";
        v.push_back(x[i]);
    }
    cout<<endl;
    cout<<count( v.begin(), v.end(),20);
    return 0;
}

```

Алгоритми find ва find_if

Алгоритмҳои оилаи find барои ҷустуҷӯ ва пайдо кардани элемент дар дохили пайдарпайи истифода бурда мешаванд.

Алгоритми find барои ҷустуҷӯи қимати value дар дохили пайдарпайи истифода бурда мешавад. Қолаби умумии он намуди зеринро дорад:

```
template <class In, class T>
    In find (In first, In last, const &value);
```

Алгоритми find_if барои ҷустуҷӯи қимати додашудаи предикати pred истифода бурда мешавад. Трази истифодабарии он шакли зеринро дорад:

```
template <class In, class Pred>
    In find (In first, In last, pred pred);
```

Ин алгоритмҳо ба элементҳои аввалии ёфташуда итератор бармегардонанд, агар амали ҷустуҷӯ бо муваффақият анҷом пазирад. **Мисол.**

```
#include <iostream>
#include <algorithm.h>
#include <map.h>
#include <vector.h>
using namespace std;

class MyClass{
public:
    bool operator()(int x){
        return x>=10 && x<=99;
    }
};

int main(int argc, char* argv[]){
    int x[10]={5,20,33,56,20,23,20,17,108, 3};
    vector <int> v;
    for (int i=0; i<10; i++) {
        cout<<x[i]<<" ";
        v.push_back(x[i]);
    }
    cout<<endl;
    cout<<*find(v.begin() ,v.end() ,33)<<endl;
    cout<<*find_if(v.begin(), v.end(), MyClass());
    return 0;
}
```

Натиҷа:

5,20,33,56,20,23,20,17,108, 3

33

20

Алгоритми adjacent_find

Алгоритми adjacent_find барои пайдо кардани ҷуфти қиматҳои ҳамсоя дар дохили пайдарпайи ба қор меравад. Қолаби умумии он намуди зеринро дорад:

```
template<class For> For adjacent_find(For first, For last);
```

```
template<class For, class BinPred>
```

```
For adjacent_find(For first, For last, BinPred pred);
```

Қолаби якуми алгоритм барои пайдо кардани ҷуфти қиматҳои якхела дар пайдарпайи истифода бурда мешавад. Агар амали ҷустуҷӯ бо муваффақият анҷом пазирад, ба нахустин элементҳои пайдарпайи итератор бардонида мешавад, дар ҳолати ақс охири пайдарпайи бармегардонида мешавад.

Қолаби дуввум бошад, ҷуфти элементҳоеро бармегардонад, ки шарти предикати pred-ро қаноат мекунонанд.

Мисол. Пайдо намудани нахустин ҷуфти элементҳои монанд.

```
#include <iostream>
#include <algorithm.h>
#include <map.h>
#include <vector.h>
using namespace std;

int main(int argc, char* argv[])
{
    int Arr[]={10,20,30,30,10,40,40,10};
    int len= sizeof(Arr)/sizeof(Arr[0]);
    vector<int> v(Arr,Arr+len);
    vector<int>::iterator it;
    it = adjacent_find(v.begin(),v.end());
    if (it!=v.end()) cout<<*it;
    else cout<<"No elements";
    return 0;
}
```

Натиҷа: 30

Мисоли дигар. Дар ин мисол ба сифати аргументи сеюм функсияи мантиқӣ истифода шудааст. Ин мисол ҳамаи ҷуфти элементҳои контейнерро мебарорад.

```
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;
bool myfunc(const int i,const int j)
{
    return i==j;
}
int main(int argc, char* argv[])
{
    int Arr[]={10,20,30,30,10,40,40,10};
    int len= sizeof(Arr)/sizeof(Arr[0]);
    vector<int> v(Arr,Arr+len);
    vector<int>::iterator it;
    it = v.begin();
    int count = 0;
    while(it!=v.end())
    {
```

```

        it = adjacent_find(++it,v.end(),myfunc);
        if (it!=v.end())
        {
            count++;
            cout<<*it<<" ";
        }
        else
            if (0==count) cout<<"No elements";
    }
    return 0;
}

```

Натиҷа: 30 40

Алгоритми search ва search_n

Алгоритми search дар дохили як пайдарпайи пайдарпайии дигарро бо таҳлили предикат ва ё бе предикат ҷустуҷӯ карда, ба элементҳои мувофиқ итератор бармегардонад. Қолаби умумии он намуди зеринро дорад:

```
Template <class For1, class For2>
```

```
For1 search (For1 first1, For1 last1, For1 first2, For1 last2);
```

```
Template <class For1, class For2, class BinPred>
```

```
For1 search (For1 first1, For1 last1,
            For1 first2, For1 last2, BinPred Pred);
```

Дар ҳолати номуваффақ будани амали ҷустуҷӯ last1-ро бармегардонад.

Алгоритми search_n барои дар дохили пайдарпайи ҷустуҷӯ намудани зерпайдарпайии дорои n қимати value истифода мешавад. Ин алгоритм низ метавонад, бе предикат ва ё бо предикат кор кунад. Қолаби умумии он намуди зеринро дорад:

```
Template <class For1, class Size, class T>
```

```
For1 search_n (For1 first1, For1 last1, Size count, const t &value);
```

```
Template <class For1, class Size, class T, class BinPred >
```

```
For1 search_n (For1 first1, For1 last1,
              Size count, const t &value, class BinPred Pred);
```

Алгоритми merge

Алгоритми merge барои ба таври баракс мураттаб кардани элементҳои пайдарпайи истифода мешавад. Қолаби умумии он намуди зеринро дорад.

```
template <class In1, class In2, class Out>
```

```
Out merge (In1 first1, In1 last1, In2 first2, In1 last2, Out Result);
```

```
template <class In1, class In2, class Out, class Compare>
```

```
Out merge (In1 first1, In1 last1,
          In2 first2, In1 last2, Out Result, Compare comp);
```

Мисол.

```
#include <iostream>
#include <algorithm>
```

```

#include <vector>
using namespace std;
int main(int argc, char* argv[])
{
    const int m=7, n=8;
    int a[n]={1,3,5,7,9,11,13,15};
    int b[m]={2,4,6,8,10,12,14};
    int c[m+n];
    merge(a,a+n, b,b+m, c);
    for (int i=0; i<(n+m); i++)
        cout<<c[i]<<" ";
    return 0;
}

```

Натиҷа: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Алгоритми next_permutation()

Алгоритми next_permutation() барои ҷойгиркунии (переста-новка) ҳамаи ҳолатҳои имконпазир элементҳои котејнер истифода бурда мешавад. Ҷойивазкуни аз n элемент ин баробар аст ба $n!$, яъне барои ҷойивазкунии n элементи додашуда n ҳолати имконпазир вучуд дорад. Мисол, бигзор тағйирёбандаи s аз навъи `char` буда, дорои 3 элемент бошад, пас, барои ҷойивазкунии элементҳои он $3!=6$ ҳолати имконпазир вучуд дорад, яъне

$S="abc"$

Ҳолатҳои имконпазир: abc, acb, bac, bca, cab, cba

Қолаби умумии алгоритми next_permutation намуди зеринро дорад:

```

template <class Bi >
    bool next_permutation (Bi first, Bi last);
template <class Bi, class Compare>
    bool next_permutation (Bi, Bi last, Compare comp);

```

Акнун ба ҳамин монанд тавассути алгоритми next_permutation() мисолро дида мебароем.

```

#include <iostream>
#include <algorithm>
#include <vector>
using namespace std;
int main(int argc, char* argv[]){
    char s[]="abcd" ;
    do{
        cout <<s<<" ";
    }while (next_permutation(s, s+4));
    cout<<endl;

    return 0;
}

```

Натиҷаи барнома намуди зеринро дорад:

```

abcd abdc acbd acdb adbc adcb bacd badc bcad bcda bdac bdca cabd cadb
cbad cbda cdab cdba dabc dacb dbac dbca dcab dcba

```

Алгоритми reverse()

Алгоритми reverse() пайдарпайи додасударо дар фосилаи додасуда, баракс мураттаб месозад. Наҳви умумии алгоритми мазкур намуди зерин аст:

```
Template <class BiIter> void reverse (Biter start, Biter end);
```

Мисол:

```
#include <iostream>
#include <algorithm>
#include <vector>
using namespace std;
#define n 20
int main(int argc, char* argv[]){
    vector <int> v;
    int i;
    cout<<" Elementhoi vectir pesh az davati function-I
reverse:"<<endl;
    for (i=1;i<=n;i++)
        v.push_back(2*i);
    for (i=0;i<v.size(); i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    cout<<"Elementhoi vectir pas az davati function-I
reverse:"<<endl;
    reverse (v.begin() , v.end() );
    for (i=0;i<v.size(); i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    return 0;
}
```

Дар ин ҷо тавре ки дида мешавад, дар аввал бо номи v векторе муаррифӣ гардида, сипас, бо он 20 адади аввалаи бутун бахшида мешавад. Баъд аз амали бахшиш, элементҳои вектор ба ҷоп бароварда шуда, функсияи reverse() даъват карда шудааст. Пас аз даъвати функсияи мазкур, элементҳои вектор аз сари нав ба ҷоп бароварда мешаванд. Натиҷаи барнома намуди зеринро дорад:

```
Elementhoi vectir pesh az davati function-I reverse:
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40
Elementhoi vectir pas az davati function-I reverse:
40 38 36 34 32 30 28 26 24 22 20 18 16 14 12 10 8 6 4 2
```

Алгоритми sort

Барои мураттаб намудани элементҳои контейнерҳо алгоритми sort истифода бурда шавад. Алгоритми мазкур дар ду намуд истифода бурда мешавад:

1. void sort (RandomAccessInIter first, RandomAccess last);
2. void sort (RandomAccessInIter first, RandomAccess last, compare comp)

Қолаби якум фосилаи додашудаи элементҳои контейнерро мураттаб мекунад ва қолаби дуюм бошад, элементҳои фосилаи нишондошударо аз рӯи шарти гузошташуда мураттаб месозад. **Мисол:**

```
#include <iostream>
#include <algorithm>
#include <ctype>
#include <vector>
using namespace std;
#define n 20
int main(int argc, char* argv[])
{
    vector <int> v;
    int i;
    for (i=0;i<n;i++)
        v.push_back(random(50));
    cout<<"Elementhoi vector:"<<endl;
    for (i=0; i<v.size() ; i++)
        cout<<v[i]<<" ";
    cout<<"\nElementhoi vector pas az murattabkuni:"<<endl;
    sort(v.begin() , v.end() ) ;
    for (i=0; i<v.size() ; i++)
        cout<<v[i]<<" ";
    cout<<endl;
    return 0;
}
```

Натиҷаи барнома намуди зеринро дорад:

```
Elementhoi vector:
49 4 36 44 12 33 6 34 36 11 11 44 0 32 43 29 13 48 6 45
Elementhoi vector pas az murattabkuni:
0 4 6 6 11 11 12 13 29 32 33 34 36 36 43 44 44 45 48 49
```

Алгоритми swap()

Алгоритми swap() барои ҷойи ду адад (қимати тағйирёбанда)-ро иваз намудан, истифода бурда мешавад. Қолаби умумии он намуди зеринро дорад:

```
template <class T> void swap (T& a, T& b);
```

Ҳоло истифодаи алгоритми swap-ро дар мисоли зерин дида мебароем.

Мисол. Барномае нависед, ки КТУ-и ду адади додашударо ҳисоб кунад.

Алгоритми remove_copy() ва алгоритми replace_copy()

Алгоритми remove() яке дигар аз алгоритмҳои мебошад, ки истифодаи зиёдтар дорад. Қолаби умумии ин алгоритм намуди зеринро дорад:

```
Template <class ForIter, class OutIter, class T> OutIter remove_copy(InIter
Start, InIter end, OutIter result, const T &val);
```

Алгоритми remove_copy() дар фосилаи додашудаи пайдарпайи қимати val –ро нест намуда, элементҳои боқимондаи ба параметри result нусхабардорӣ мекунад.

Алгоритми `replace_copy()` низ ба алгоритми `remove_copy()` буда, танҳо он элементи нишондодашударо ба элементи додашуда иваз ме-кунад. Қолаби умумии ин алгоритм намуди зеринро дорад:

Template <class ForIter, class OutIter, class T> OutIter `replace_copy()`InIter Start, InIter end, OutIter result, const T &old, const T &new);

Мисол. Барномае нависед, ки дар сатри додашуда аввал ҳарфи а-и сатри додашударо нест намуда, сипас, аз нав онро бо и иваз намояд.

```
#include <iostream>
#include <algorithm>
#include <ctype>
#include <vector>
using namespace std;
int main(int argc, char* argv[]){
    char str[]="Classi algorithm behtarin class meboshad!!";
    int i;
    vector <char> v, v1(100);
    cout<<"satri dodashuda:"<<endl;
    for (i=0;str[i];i++)
        v.push_back(str[i]);
    for (i=0;v[i]; i++)
        cout<<v[i];
    cout<<endl;
    remove_copy(v.begin() , v.end(), v1.begin(), 'a' );
    cout<<"satr pas az nest kardani harfi a:"<<endl;
    for (i=0;v1[i]; i++)
        cout<<v1[i];
    cout<<endl<<endl;
    replace_copy(v.begin() , v.end(), v1.begin() , 'a', 'u');
    cout<<"satr pas az ivaz kardani harfi a ba u:"<<endl;
    for (i=0;v1[i]; i++)
        cout<<v1[i];
    cout<<endl<<endl;
    return 0;
}
```

Дар ин ҷо аввал бо номи `str` сатре аз навъи `char` муаррифӣ гар-дидааст. Баъд элементҳои сатр бо вектори символӣ бахшида шуда, ба ҷоп бароварда шудаанд. Барои ҳарфи а-ро нест кардан функсияи `remove_copy()` ба намуди зерин даъват карда шудааст

```
remove_copy(v.begin() , v.end(), v1.begin(), 'a' );
```

пас аз даъвати ин функсия, ҳарфҳои а-и вектори `v` нест гардида, сим-волҳои боқимондаи он ба вектори `v1` бахшида мешаванд. Барои ҳарфҳои а-и вектори `v1`- ро ба и иваз кардан аз функсияи `replace_copy()` ба сурати зерин истифода карда мешавад:

```
replace_copy(v.begin() , v.end(), v1.begin() , 'a', 'u');
```

Натиҷаи барнома намуди зеринро дорад:

```
satri dodashuda:
Classi algorithm behtarin class meboshad!!
```



```
satr pas az nest kardani harfi a:  
Classi lgorithm behtrin class meboshd!!
```

```
satr pas az ivaz kardani harfi a ba u:  
Clussi ulgorithm behturin class meboshud!!
```

Алгоритми transform()

Алгоритми transform яке аз алгоритмҳои дигаре мебошад, ки он барои табдилдиҳии пайдарпайи хизмат мекунад. Алгоритми мазкур дар ду қолаб истифода бурда мешавад:

1. Template <class InIter, class OutIter, class func>

```
OutIter transform(InIter Start, InIter end, OutIter result,  
func binaryfunc);
```

2. Template <class InIter1, class InIter2, class OutIter, class func>

```
OutIter transform(InIter1 Start1, InIter1 end1, InIter2 Start12,  
OutIter result, func binaryfunc);
```

Алгоритми transform() фосилаи [start, end] –ро табдил дода, ба параметри result нусхабардорӣ мекунад. Дар қолаби якум фосила параметри [start, end] дода шудааст. Барои табдилдиҳи ба функсия параметри unaryfunc дода мешавад. Функсия аргументеро ба сифати қимат қабул намуда, онро табдил дода бармегардонад. Қолаби дуюм бошад, барои табдилдиҳи аз функсияи бинарӣ истифода мебаранд, ки ба сифати аргументи аввала элемент аз пайдарпайии якум ва ба сифати аргументи дуввум элемент аз пайдарпайии дуюм қабул мекунад.

Дар мисоли зерин функсияи func() барои ҳарфи хурдро ба калон табдил додан истифода бурда мешавад.

```
#include <iostream>  
#include <algorithm>  
#include <ctype>  
#include <vector>  
using namespace std;  
char func(char s)  
{  
    return s+32;  
}  
int main(int argc, char* argv[]){  
    char str[]="ABCDEFGHJKLMNOPRSTU";  
    int i;  
    vector <char> v, v1(100);  
    cout<<"satri dodashuda:"<<endl;  
    for (i=0;str[i];i++)  
        v.push_back(str[i]);  
    for (i=0;v[i]; i++)  
        cout<<v[i];  
    cout<<endl;  
    vector <char>::iterator p=v.begin();  
    p=transform(v.begin(), v.end(), v.begin(), func);
```

```

cout<<"Satr pas az tagirdihi:"<<endl;
for (i=0;v[i]; i++)
    cout<<v[i];
return 0;
}

```

Ин барнома ҳарфҳои хурди сатри додашударо ба бузург табдил медиҳад. Натиҷаи барнома намуди зеринро доранд:

```

satri dodashuda:
ABCDEFGHIJKLMNPRSTU
Satr pas az tagirdihi:
abcdefghijklmoprstu

```

Алгоритми random_shuffle()

Алгоритми мазкур барои ба таври тасодуфӣ ҷойҳои элементҳои контейнерро иваз намудан, истифода бурда мешавад. Қолаби умумии ин алгоритм намуди зеринро дорад:

```
void random_shuffle (RandomAccessIter first, RandomAccess last);
```

Мисол:

```

#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
using namespace std;
#define n 15
int main(int argc, char* argv[])
{
    vector <int> v;
    int i;
    for (i=0;i<n;i++)
        v.push_back(random(50));
    cout<<"Elementhoi vector:"<<endl;
    for (i=0; i<v.size() ; i++)
        cout<<v[i]<<" ";
    cout<<"Elementhoi vector pas az ivazkunii joi onho: "<<endl;
    random_shuffle(v.begin() , v.end() ) ;
    for (i=0; i<v.size() ; i++)
        cout<<v[i]<<" ";
    cout<<endl;
    return 0;
}

```

Натиҷаи барнома:

```

Elementhoi vector:
49 4 36 44 12 33 6 34 36 11 11 44 0 32 43 29 13 48 6 45
Elementhoi vector pas az ivazkunii joi onho:

11 44 29 6 36 33 49 0 13 6 12 48 43 32 44 4 45 36 11 34

```

Алгоритми accumulate ()

Алгоритми accumulate() барои ҳисоб намудани суммаи элементҳои контейнер дар фосилаи додашуда истифода бурда мешавад. Алгоритми мазкур аз библиотекаи <algorithm> набуда, балки он аз библиотекаи <numeric.h> мебошад. **Мисол:**

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;
#define n 15
int main(int argc, char* argv[]){
    vector <int> v;
    int i;
    for (i=0;i<n;i++)
        v.push_back(random(50));
    cout<<"Elementhoi vector:"<<endl;
    for (i=0; i<v.size() ; i++)
        cout<<v[i]<<" " ;
    cout<<endl;
    double sum=accumulate(v.begin() , v.end() , 0.0);
    cout<<"sum="<<sum<<endl;
    return 0;
}
```

Натиҷаи барнома шакли зеринро дорад:

```
Elementhoi vector:
49 4 36 44 12 33 6 34 36 11 11 44 0 32 43 29 13 48 6 45
sum=536
```

Алгоритмҳои copy() ва copy_backward()

Алгоритмҳои copy() ва copy_backward() барои як векторро ба вектори дигар нусхабардорӣ кардан, истифода бурда мешаванд.

Қолаби умумии алгоритми copy () намуди зеринро дорад:

```
template <class In, class Out>
    Out copy (Inp first, In last, Out result);
```

Қолаби умумии алгоритми copy_backward() бошад, намуди зеринро дорад:

```
template <class Bi1, class Bi2>
    Bi2 copy_backward (Bi 1 first, Bi 1 last, Bi 2 result);
```

Мисол:

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;
```

```

int main(int argc, char* argv[]){
    int x[5]={1,2,3,4,5};
    int y[5]={6,7,8,9,10};
    vector <int> v1(x, x+5), v2(y, y+5), v3(y, y+5);
    cout << " Vectory yakum v1"<<endl;
    for (int i=0; i<v1.size(); i++)
        cout<<v1[i]<<" " ;
    cout<<endl;
    cout << " Vectory duvum v2"<<endl;
    for (int i=0; i<v2.size(); i++)
        cout<<v2[i]<<" " ;
    cout<<endl;
    cout << " Vectory seyum v3"<<endl;
    for (int i=0; i<v3.size(); i++)
        cout<<v3[i]<<" " ;
    cout<<endl;
    cout << " Nuskhabori kardani vectory yakum v1 ba duvum
v2"<<endl;
    copy( v1.begin(), v1.end(),v2.begin());
    for (int i=0; i<v2.size(); i++)
        cout<<v2[i]<<" " ;
    cout<<endl;
    cout << "Nuskhabori kardani vectory duvum v2 ba seyum
v3"<<endl;
    copy_backward( v2.begin(), v2.end(),v3.begin() );
    for (int i=0; i<v3.size(); i++)
        cout<<v3[i]<<" " ;
    return 0;
}

```

Дар ин мисол Тавре ки дида мешавад, ду массив бо номи x ва y, ки андозаашон ба 5 баробар аст, муаррифӣ шудаанд. Сипас, се вектор бо номҳои v1, v2 ва v3 муаррифӣ шуда, ба сифати қимати вектори аввала, қиматҳои массиви якум ва ба сифати ду вектори дигар, қимати массиви дуюм бахшида шудааст.

Дар даъвати алгоритми copy() ҳамаи элементҳои вектори якум ба вектори дувум нусхабардорӣ шудаанд. Даъвати алгоритми copy_backward() бошад, ҳамаи элементҳои вектори якумро ба сеюм нусхабардорӣ мекунад. Дар ин даъват параметри сеюм, ки ба охири контейнер ишора мекунад, барои аз охир ба аввал нусхабардорӣ кардани элементҳои вектор истифода мешавад. Натиҷаи барнома намуди зеринро дорад:

```

Vectory yakum v1
1 2 3 4 5
Vectory duvum v2
6 7 8 9 10
Vectory seyum v3
6 7 8 9 10
Nuskhabori kardani vectory yakum v1 ba duvum v2
1 2 3 4 5

```

Nuskhabardori kardani vectory duyum v2 ba seyum v3
6 7 8 9 10

Алгоритмҳои стандартӣ болои ҳама намуд контейнерҳо амал мекунанд. Дар поён истифодаи алгоритми `copy()`-ро барои нусха-бардори намудани элементҳои вектор ба `List` оварда шудааст.

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;

int main(int argc, char* argv[]){
    int x[5]={1,2,3,4,5};
    vector <int> v(x, x+5);
    list <int> L, L1, L2;
    copy(v.begin() , v.end() , back_inserter(L));
    cout<<"Guzoshtani elementhoi vector dar ruyhat "<<endl;
    list <int>::iterator p;
    p=L.begin() ;
    while (p!=L.end() ){
        cout<<*p<<" ";
        p++;
    }
    cout<<endl;
    cout<<"Guzoshtani elementhoi vector dar ruyhat bo ravishi
digar"<<endl;
    copy(v.begin() , v.end() , inserter(L2, L2.begin() ));
    p=L2.begin() ;
    while (p!=L2.end() ){
        cout<<*p<<" ";
        p++;
    }
    cout<<endl;
    cout<<"Guzoshtani elementhoi vector dar ruyhat (baraks)"<<endl;
    copy(v.begin() , v.end() , front_inserter(L1));
    p=L1.begin() ;
    while (p!=L1.end() )
    {
        cout<<*p<<" ";
        p++;
    }
    return 0;
}
```

Дар ин мисол аввал як массив ва се `list` аз навъи бутун муаррифӣ гардидаанд. Дар даъвати якум ва дуҷуми алгоритми `copy()` элементҳои вектор ба `List`-ҳои якум ва дуҷум нусхабардорӣ шудаанд. Дар даъвати сеюм бошад, элементҳои вектор ба `list`-и `L3` бо тартиби баръакс нусхабардорӣ шудаанд. Пас

аз ҳар даъват элементҳои List дар саҳифаи намоиш бароварда мешаванд. Натиҷаи барнома намуди зеринро дорад:

```
Guzoshtani elementhoi vector dar ruyhat
1 2 3 4 5
Guzoshtani elementhoi vector dar ruyhat bo ravishi digar
1 2 3 4 5
Guzoshtani elementhoi vector dar ruyhat (baraks)
5 4 3 2 1
```

Алгоритми `replace`, `replace_if`, `replace_copy` ва `replace_copy_if`

Алгоритмҳои оилаи `replace` барои иваз намудани қимати элементи додашударо бо қимати нав ё предикат хизмат мекунанд. Қолаби умумии алгоритмҳои ин оила намуди зеринро дорад:

```
template <class For, class T>
void replace (For first, For last,
              const T &old_value, const T &new_value);
template <class For, class Pred, class T>
void replace_if (For first, For last,
                 Pred pred, const T &new_value);
template <class In, class Out, class T>
Out replace_copy (In first, In last, Out result,
                  const T &oldvalue, const T &newvalue);
template <class Iterator, class Out, class Pred, class T>
Out replace_copy_if (Iterator first, Iterator last, Out result,
                    Pred pred, const T &newvalue);
```

Мисолеро дида мебароем, ки дар он вектори `a` бо вектори `b` бо истифода аз итератори гузориш ва иваз кардани ҳамаи элементҳои тоқ ба адади 20 нусхабардорӣ карда мешавад.

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;

bool odd(int x){
    return x%2!=0;
}
int main(int argc, char* argv[])
{
    vector <int> a, v;
    vector<int>::iterator i;

    for (int k=1; k<=10; k++)
        v.push_back(k);
    for (i=v.begin(); i!=v.end();i++)
        cout<<*i<<" ";
```

```

    cout<<endl;
    replace_copy_if(v.begin(), v.end(), inserter(a, a.begin()), odd,
20);
    for (i=a.begin(); i!=a.end();i++)
        cout<<*i<<" ";
    return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

1 2 3 4 5 6 7 8 9 10
20 1 20 4 20 6 20 8 20 10

```

Алгоритми generate ва generate_n

Алгоритми generate барои ба натиҷаи амалиёт иваз намудани ҳамаи элементҳои контейнер истифода бурда мешавад. Ин имконият фароҳам меорад, ки контейнерро аз қиматҳои ягона (уникалӣ) пур кунем. Қолаби умумии он намуди зеринро дорад:

```

template <class For, class Generator>
    void generate (For first, For last, Generator gen);
template <class Out, class Size, class Generator>
    void generate _n(Out first, Size n, Generator gen);

```

Мисол. Истифодаи алгоритмҳои generate ва generate_n.

```

#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
#include <cmath>
using namespace std;

int func(){
    static x=1;
    return pow(++x,2);
}
int main(int argc, char* argv[])
{
    int a[10], i;
    generate( a,a+10, func);
    for (i=0; i<10; i++)
        cout<<a[i]<<" ";
    return 0;
}

```

Натиҷаи барнома намуди зеринро дорад:

```

4 9 16 25 36 49 64 81 100 121

```

Алгоритми fill ва fill_n

Алгоритми fill барои ба қимати value иваз намудани ҳамаи элементҳои пайдарпай, ки онҳоро итераторҳои first ва last муайян кардаанд, хизмат

мекунад. Алгоритми fill_n бошад, ин амалро танҳо барои n элемент иҷро мекунад. Қолаби умумии он намуди зеринро дорад:

```
template <class For, class T>
void fill (For first, For last, const T &value);
template <class Out, class Size , class T>
void fill_n (Out first, Size n, const T &value);
```

Мисол.

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
#include <cmath>
using namespace std;

int main(int argc, char* argv[])
{
    int a[10], i;
    fill(a,a+10, 2);
    for (i=0; i<10; i++)
        cout<<a[i]<<" "; // 2 2 2 2 2 2 2 2 2 2
    cout<<endl;
    fill_n(a+4, 5, 3);
    for (i=0; i<10; i++)
        cout<<a[i]<<" "; // 2 2 2 2 3 3 3 3 3 2
    return 0;
}
```

Алгоритми partial_sort ва partial_sort_copy

Алгоритми partial_sort барои қисман мурағтаб намудани элементҳои массив ё вектор истифода бурда мешавад. Баъд аз иҷрои ин алгоритм элементҳои фосилаи аз first to last ҳолатеро соҳиб мешаванд, ки ҳангоми пурра мурағтаб намудани массив онро аз даст намедиханд.

Алгоритми partial_sort_copy айнан амали алгоритми partial_sort бо нусхабардорӣ намудани элементҳои пайдарпай иҷро мекунад.

Қолаби умумии ин алгоритмҳо намуди зеринро дорад:

```
template <class Ran>
void partial_sort(Ran first, Ran middle, Ran last);
template <class Ran, class Compare>
void partial_sort(Ran first, Ran middle, Ran last, Compare Com);
template <class In, class Ran>
Ran partial_sort_copy(In first, In last, Ran result_first, Ran result_last);
template <class In, class Ran, class Compare >
Ran partial_sort_copy(In first, In last,
Ran result_first, Ran result_last, class Compare);
```


Мисол.

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;

int main(int argc, char* argv[])
{
    int a[10], i;
    cout<<"Elementhoi massiv pesh az partial_sort:";
    for (i=0; i<10; i++) {
        a[i]=rand()%25;
        cout<<a[i]<<" ";
    }
    vector <int> v(a,a+10);
    vector<int>::iterator ii=v.begin()+2;
    cout<<endl;
    partial_sort(v.begin(), v.begin()+5, v.end()) ;
    cout<<"Elementhoi massiv pas az partial_sort:";
    for (i=0; i<10; i++)
        cout<<v[i]<<" ";
    cout<<endl;
    return 0;
}
```

Натиҷаи барнома:

```
Elementhoi massiv pesh az partial_sort:
5 7 15 6 17 20 15 23 1 4
Elementhoi massiv pas az partial_sort:
1 4 5 6 7 20 17 23 15 15
```

Алгоритми nth_element

Алгоритми nth_element низ барои қисман мурағтаб намудани элементҳои пайдарпайи истифода бурда мешавад. Пас аз даъвати алгоритми мазкур, қимати элементҳои дар итератори nth қарордошта, ҳолатеро соҳиб мешаванд, ки ҳангоми пурра мурағтаб намудани массив онро аз даст намудихад, яъне дар тарафи чапи он элементҳои аз он хурд ва дар тарафи ростии он элементҳои аз он калон ҷойгир мешаванд.

Қолаби умумии ин алгоритм намуди зеринро дорад:

```
template <class Ran>
void nth_element(Ran first, Ran nth, Ran last);
template <class Ran, class Compare>
void nth_element (Ran first, Ran nth, Ran last, Compare Com);
```

Мисол.

```
#include <iostream>
#include <algorithm>
#include <ctype.h>
```

```

#include <vector>
#include <numeric>
using namespace std;

int main(int argc, char* argv[])
{
    int a[10], i;
    cout<<"Elementhoi massiv pesh az nth_element:"<<endl;
    for (i=0; i<10; i++) {
        a[i]=rand()%35;
        cout<<a[i]<<" ";
    }
    vector <int> v(a,a+10);
    vector<int>::iterator ii=v.begin() +6;
    cout<<endl;
    nth_element(v.begin(), ii, v.end()) ;
    cout<<"Elementhoi massiv pas az nth_element:"<<endl;
    for (i=0; i<10; i++)
        cout<<v[i]<<" ";
    cout<<endl;
    return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:

```

Elementhoi massiv pesh az nth_element:
25 27 5 1 12 25 10 23 11 9
Elementhoi massiv pas az nth_element:
1 5 10 9 11 12 23 25 27 25

```

Алгоритмҳои кор бо маҷмӯъҳо

Маҷмӯъ гуфта, гурӯҳи элементҳои яҷинсаро меноманд. Дар забони C++ контейнерҳои зиёде барои кор кардан ба гурӯҳи элементҳо мавҷуданд. Илова бар ин дар STL барои кор кардан бо маҷмӯъҳо алгоритмҳои махсусе мавҷуданд, ки ҳангоми истифодаи онҳо маҷмӯъҳо ҳамчун массивҳо ё вектор муаррифӣ карда мешаванд.

Дар риёзи барои ишора кардани маҷмӯъҳо аз ҳарфҳои калони латини истифода бурда шуда, элементҳои маҷмӯъ дар дохили қавси фигуравӣ навишта мешаванд. Барои бо маҷмӯъҳо амалҳои зеринро иҷро кардан алгоритм мавҷуд аст.

Чамъ;

Зарб;

Фарқ;

Фарқи симметрӣ.

- 1) Чамъи ду маҷмӯъ гуфта, маҷмӯъеро меноманд, ки элементҳои он аз якҷоякунии маҷмӯи якум ва дуюм ташкил ёфтаанд.
- 2) Зарби ду маҷмӯъ гуфта, маҷмӯъеро меноманд, ки элементҳои он аз элементҳои якхелаи маҷмӯи якум ва дуюм ташкил ёфтаанд.

- 3) Фарқи ду маҷмӯъ гуфта, маҷмӯъеро меноманд, ки элементҳои он аз он элементҳои маҷмӯи якум, ки дар маҷмӯи дуюм вучуд надоранд ташкил ёфтаанд.
- 4) Фарқи симметрии ду маҷмӯъ гуфта, маҷмӯъеро меноманд, ки элементҳои он аз элементҳои гуногуни маҷмӯи якум ва дуюм ташкил ёфтаанд, яъне он элементҳои маҷмӯи якум, ки дар маҷмӯи дуввум нестанд ва баракс.

Мисол. Бигзор ду маҷмӯъ додашуда бошанд:

$$S1 = [1, 3, 5, 6, 8, 9];$$

$$S2 = [3, 6, 8, 11, 15].$$

Пас, иҷрои амалҳои арифметикӣ намуди зеринро дорад:

$$S = S1 + S2 = [1, 3, 5, 6, 8, 9, 11, 15];$$

$$P = S1 * S2 = [3, 6, 8];$$

$$F = S1 - S2 = [1, 5, 9].$$

Бояд қайд кард, ки барои амали тарҳ дар маҷмӯъҳо амали ҷойивазкуни ҷой надорад, яъне

$$S1 - S2 \neq S2 - S1$$

Алгоритми includes

Алгоритми includes барои санҷиши мавҷудияти як пайдарпайӣ (маҷмӯъ) дар дохили пайдарпайии (маҷмӯи) дигар истифода бурда мешавад. Агар ҳамаи элементҳои пайдарпайии [first2, last2) дар пайдарпайии [first1, last1) мавҷуд бошанд, қимати ҳақ бармегардонад. Қолаби умумии ин алгоритм намуди зеринро дорад:

```
template <class In1, class In2>
```

```
bool includes ( In1 first1, In1 last1, In2 first2, In2 last2 );
```

```
template <class In1, class In2, class Compare>
```

```
bool includes ( In1 first1, In1 last1, In2 first2, In2 last2,
```

```
Compare comp );
```

Мисол.

```
#include <iostream>
#include <algorithm>
#include <vector>
using namespace std;

bool myfunction (int i, int j){
    return i<j;
}

int main(int argc, char* argv[])
{
    int x[] = {5,10,15,20,25,30,35,40,45,50};
    int y[] = {40,30,20,10};
    sort (x,x+10);
    sort (y,y+4);
    if (includes(x,x+10,y,y+4) )
        cout << "y dar x mekhobad!\n";
    if (includes(x,x+10,y,y+4, myfunction) )
```

```

        cout << "y dar x mekhobad!\n";
    return 0;
}

```

Натиҷаи барнома:
y dar x mekhobad!
y dar x mekhobad!

Алгоритми set_intersection

Алгоритми set_intersection аз зарби ду маҷмӯро бо намуди ғайримураттаб маҷмӯи нав месозад. Қолаби умумии он намуди зеринро дорад:

```
template <class In1, class In2, class Out>
```

```
    Out set_intersection (In1 first1, In1 last1, In2 first2, In2 last2, Out result);
```

```
template <class In1, class In2, class Out, class Compare>
```

```
    Out set_intersection (In1 first1, In1 last1, In2 first2, In2 last2,
        Out result, Compare comp);
```

Мисол.

```

#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;

int main(int argc, char* argv[])
{
    vector<int> v(10);
    vector<int>::iterator it;
    int x[] = {1,3,5,6,8,9};
    int y[] = {3,6,8,11,15};  sort (x,x+10);
    sort (y,y+4);
    it=set_intersection (x, x+10, y, y+6, v.begin());
    v.resize(it-v.begin());
    cout << "majmuho doroi " << (v.size()) << " elementi monand
meboshand:\n";
    for (it=v.begin(); it!=v.end(); ++it)
        cout << " " << *it;
    cout << '\n';
    return 0;
}

```

Натиҷаи барнома:
majmuho doroi 3 elementi monand meboshand:
3 6 8

Алгоритми set_difference, set_symmetric_difference

Алгоритми set_difference он элементҳоро, ки аз ду пайдарпайии додашуда танҳо дар якумаш меҳобанд дар Out нусхабардорӣ мекунад, яъне

фарқи ду маҷмӯи додашуда маҷмӯи нав месозад. Қолаби умумии он намуди зеринро дорад:

```
template <class In1, class In2, class Out>
    Out set_difference (In1 first1, In1 last1, In2 first2, In2 last2,
        Out result);
template <class In1, class In2, class Out, class Compare>
    Out set_difference (In1 first1, In1 last1, In2 first2, In2 last2, Out
        result, Compare comp);
```

Алгоритми `set_symmetric_difference` бошад, он элементҳоеро, ки аз ду пайдарпайии додашуда танҳо дар якеаш меҳобанд дар `Out` нусхабардорӣ мекунад, яъне он аз фарқи симметрии ду маҷмӯъ маҷмӯъи нав месозад. Қолаби умумии он намуди зеринро дорад:

```
template <class In1, class In2, class Out>
    Out set_symmetric_difference (In1 first1, In1 last1, In2 first2,
        In2 last2, Out result);
template <class In1, class In2, class Out, class Compare>
    Out set_symmetric_difference (In1 first1, In1 last1, In2 first2,
        In2 last2, Out result, Compare comp);
```

Мисоли алгоритми `set_difference`.

```
#include <iostream>
#include <algorithm>
#include <vector>
using namespace std;

int main(int argc, char* argv[]){
    vector<int> v(10);
    vector<int>::iterator it;
    int x[] = {1,3,5,6,8,9};
    int y[] = {3,6,8,11,15};
    sort (x,x+6);
    sort (y,y+5);
    it=set_difference (x, x+6, y, y+5, v.begin());
    v.resize(it-v.begin());
    cout << "dar majmui yakum " << (v.size()) << " element ast,ki dar
majmui duyum vujud nadorand:\n";
    for (it=v.begin(); it!=v.end(); ++it)
        cout << ' ' << *it;
    cout << '\n';
    return 0;
}
```

Натиҷаи барнома:

dar majmui yakum 3 element ast,ki dar majmui duyum vujud nadorand:

1 5 9

Мисоли алгоритми `set_symmetric_difference`.

```
#include <iostream>
#include <algorithm>
#include <vector>
```

```

using namespace std;
int main(int argc, char* argv[]){
    vector<int> v(10);
    vector<int>::iterator it;
    int x[] = {12,10,15,20,25,30,35,40,45,50};
    int y[] = {10, 25, 3, 40, 8};
    sort (x,x+10);
    sort (y,y+4);
    it=set_symmetric_difference (x, x+10, y, y+6, v.begin());
    v.resize(it-v.begin());
    cout << "dar majmui yakum va duyum " << (v.size()) << " elementi
gunogun vujud dorad:\n";
    for (it=v.begin(); it!=v.end(); ++it)
        cout << ' ' << *it;
    cout << '\n';
    return 0;
}

```

Натиҷаи барнома

dar majmui yakum va duyum 5 elementi gunogun vujud dorad:
1 5 9 11 15

Алгоритми set_union

Алгоритми set_union аз суммаи ду маҷмӯъ ба намуди ғайримураттаб маҷмӯъи нав месозад. Қолаби умумии он шакли зеринро дорад:

```
template <class In1, class In2, class Out>
```

```
    Out set_union (In1 first1, In1 last1, In2 first2, In2 last2, Out result);
```

```
template <class In1, class In2, class Out, class Compare>
```

```
    Out set_union (In1 first1, In1 last1, In2 first2,
        In2 last2, Out result, Compare comp);
```

Мисол.

```

#include <iostream>
#include <algorithm>
#include <ctype.h>
#include <vector>
#include <numeric>
using namespace std;

int main(int argc, char* argv[])
{
    vector<int> v(10);
    vector<int>::iterator it;
    int x[] = {5,10,15, 20,25, 30,35,40,45, 50};
    int y[] = {10, 20,30,40,50};
    sort (x,x+10);
    sort (y,y+4);
    it=set_union(x, x+10, y, y+6, v.begin());
    v.resize(it-v.begin());
    for (it=v.begin(); it!=v.end(); ++it)

```

```
        cout << ' ' << *it;
    cout << '\n';
    return 0;
}
```

Натиҷа:

1 3 5 6 8 9 11 15

Элементҳои маҷмӯъҳо на танҳо ададҳо, инчунин метавонанд символҳо, сатрҳо ва ғайра бошанд. Барои далели ин гуфтаҳо, мисоли дигареро тавассути алгоритми `set_union` дида мебароем, ки элементҳои маҷмӯъ дар он сатрҳо мебошанд.

```
#include <iostream>
#include <algorithm>
#include <vector>
#include <cassert>
using namespace std;

int main(int argc, char* argv[])
{
    bool result;
    string s("ABCDEFGH");
    string s2("ABICDFKL");
    vector<char> v1(s.begin(), s.end());
    vector<char> v2(s2.begin(), s2.end());
    vector<char> su;
    set_union(v1.begin(), v1.end(),v2.begin(),      v2.end(),
back_inserter(su));
    for(int i=0;i<su.size();i++){
        cout << su[i];
    }
    return 0;
}
```

Натиҷаи барнома:

ABCDEFGHICDFKL

Саволҳо барои мустаҳкамкунӣ

1. STL чист ва барои чӣ истифода бурда мешавад?
2. Контейнери классҳо чӣ маъно дорад?
3. Векторҳо чӣ гуна додаанд ва чӣ тавр муаррифӣ карда мешаванд?
4. Чӣ тавр бо элементҳои векторҳо дастрас шудан мумкин аст?
5. Итератор чист ва чӣ тавр муаррифӣ карда мешавад?
6. Тавассути итератор чӣ тавр ба элементҳои вектор дастрас шудан мумкин аст?
7. Кадом функсияҳои кор бо векторҳо медонед, мисолҳо биёред?
8. Рӯйхатҳо чӣ гуна додаанд?
9. Оё элементҳои рӯйхат метавонанд, элементҳои як массив бошанд?
10. Элементҳои як рӯйхат чӣ тавр дар саҳифаи тасвир навишта ё бароварда мешаванд?
11. Тасвирҳо чӣ гуна додаанд ва чӣ тавр муаррифӣ карда мешаванд?

12. Кадом намуди алгоритмҳо дар STL вучуд доранд?
13. Бо ёрии алгоритмҳои шуморишӣ мисолҳо биёред?
14. Аз алгоритми swap () чӣ тавр истифода бурда мешавад?
15. Бо ёрии ҳамаи контейнерҳо мисолҳо биёред.

Намунаи ҳалли мисолҳо

- 1) Дар пайдарпайии ададҳои a_1, a_2, a_3, \dots ($0 \leq a_i < 10\,000$) аъзои якум дода шудааст. Аъзоҳои боқимонда аз рӯи формулаи $a_i = (a_i - 1)^2 \bmod 10\,000$ ҳисоб карда мешаванд. Аъзои N ($1 \leq N \leq 2\,000\,000\,000$)-уми пайдарпайӣ ёфта шавад.

Доҳилкунӣ аз файли input.txt дар сатри якум адади a_1 ва N тавассути фосила навишта мешавад.

Хориҷкунӣ дар файли output.txt. Як адад - a_N хориҷ карда мешавад.

Мисолҳо

Доҳилкунӣ 1	Доҳилкунӣ 2
4 3	0 2000000000
Хориҷкунӣ 1	Хориҷкунӣ 2
256	0

Матни барнома:

```
#include <iostream>
#include <cstdio>
#include <vector>
using namespace std;
const int size = 1010;
long long n;
vector<int> mas;
vector<int> mem;
vector<int> pos;
void input()
{
    mas.resize(size);
    mem.resize(size);
    pos.resize(size);
    cin>>mas[0]>>mas[1]>>mas[2];
    mem[mas[2] + 10*mas[1] + 100*mas[0]] = 1;
    cin>>n;
    n--;
}
void solve()
{
    int base = -1;
    int len = -1;
    for (int i=3;i<size;i++)
    {
        mas[i] = (mas[i-1] + mas[i-2] + mas[i-3])%10;
        int value = mas[i-1] + 10*mas[i-2] + 100*mas[i-3];
        if (mem[value] && i!=3)
```



```

        {
            base = pos[value];
            len = i - 3 - base;
            break;
        }
        mem[value] = 1;
        pos[value] = i-3;
    }
    if (n < base + len)
        cout<<mas[n];
    else
    {
        int index = base + (n - base) % len;
        cout<<mas[index];
    }
}
int main()
{
    freopen("input.txt","r",stdin);
    freopen("output.txt","w",stdout);
    input();
    solve();
    return 0;
}

```

- 2) Дар муаммои зарб бозӣ бо сафи картаҳо карда мешавад, ки ҳар яке аз онҳо дорои адади (рақами) бутуни мусбат мебошад. Ҳангоми ход бозингар яке аз картаҳоро аз саф нест карда, миқдори холҳоро соҳиб мегардад, ки баробар аст ба ҳосили зарби адади (рақами) картаи нестшуда ва адади (рақами) картаҳои тарафи чап ва рости он. Картаҳои аввал ва охири сафро нест кардан мумкин нест. Пас аз ходи охирон танҳо ду карта боқӣ мемонад.

Мақсади бозӣ нест намудани картаҳо бо ҳамон тартибе, ки миқдори умумии холҳои наборкардашударо камтарин гардад.

Масалан, агар картаҳо дорои 10, 1, 50, 20 ва 5 бошанд, бозингар метавонад, аввал картаи рақами 1 ва баъд 20 ва 50-ро гирифта дорои холҳои зерин гардад

$$10 \cdot 1 \cdot 50 + 50 \cdot 20 \cdot 5 + 10 \cdot 50 \cdot 5 = 500 + 5000 + 2500 = 8000.$$

Агарон картаҳоро бо тартиби баракс гирифта шаванд, яъне аввал 50, баъд 20 ва баъд 1, онгоҳ миқдори холҳо ба сурати зерин ҳисоб карда мешаванд:

$$1 \cdot 50 \cdot 20 + 1 \cdot 20 \cdot 5 + 10 \cdot 1 \cdot 5 = 1000 + 100 + 50 = 1150.$$

Додаҳои вурӯдӣ

Дар сатри якуми файли input.txt миқдори картаҳо N навишта мешавад, дар сатри дуввум бошад, бо фосила N ($3 \leq N \leq 100$) рақами картаҳо дохил карда мешавад, рақами картаҳо адади бутуни аз 1 то 100 мебошад.

Додаҳои хуруҷӣ

Дар файли output.txt як адади бутун, ки миқдори минималии имконияти холҳоро ифода мекунад, бароварда мешавад.

Мисол

Доҳилкунӣ	Хориҷкунӣ
6	3650
10 1 50 50 20 5	

Матри барнома

```

#include <iostream>
#include <vector>
#include <cstdio>
using namespace std;
int n;
vector<int> mas;
vector<vector<int> > mem;
void input(){
    cin>>n;
    mas.resize(n);
    for (int i=0;i<n;i++)
        cin>>mas[i];
    mem = vector<vector<int> > (n,vector<int>(n));
}
void solve()
{
    for (int len = 2; len < n; len++) {
        for (int i=0;i<n;i++){
            int j = i + len;
            if (j == n)
                break;
            int res = 1e9;
            for (int m = i+1; m<=j-1; m++)
            {
                int cur = mem[i][m] + mas[i] * mas[m] * mas[j] +
mem[m][j];
                res = min(res,cur);
            }
            mem[i][j] = res;
        }
    }
    cout<<mem[0][n-1];
}
int main(){
    freopen("input.txt","r",stdin);
    freopen("output.txt","w",stdout);
    input();
    solve();
    return 0;
}

```

Мисолҳо барои кори мустақилона

Масъала (мисол)-ҳои гурӯҳи Array ва Matrix (Боби 6) -ро бо истифода аз имкониятҳои STL ҳал кунед.

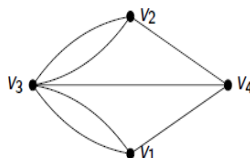
Боби 13. Графҳо

Муқаддима

Мавзӯи графҳо яке аз мавзӯҳои муҳим ва асосӣ ба ҳисоб меравад, ки он дар риёзиёти дискретӣ, физика, алгебра, техникаи ҳисоббарор, системаи коркарди маълумотҳо, равандҳои технологӣ, шабакаҳои компютерӣ, системаи таъмини барқӣ, иқтисодиёт, таҳлили занҷири Марков дар назарияи эҳтимолият, лоиҳакаиҳои схемаҳои электронӣ, ҷомеашиносӣ, системаи сохтмони роҳҳо ва ғайра истифодаи васеъ дорад.

Граф гуфта, маҷмӯи объектҳои аз ҷиҳати навъ якхеларо меноманд. Ба ин мисол шуда метавонанд: маҷмӯи шаҳрҳо, маҷмӯи одамон ва ғайра.

Назарияи гарфҳо аввалин мартиба аз масъалаи 7-пули шаҳри Кенигсберг ба вуҷуд омад. Шакли пулҳои мазкур чунин буд:



Моҳияти масъала чунин буд: Оё метавон ба 7 пулӣ шаҳри Кенигсберг ҳаракат карда, ба ҷои аввала баргашт. Дар ин муддат ба ҳар як пул бояд як бор гузашт. Ин масъаларо математикӣ швётсариягӣ Л. Эйлер бо истифода аз графҳо ҳал кардааст.

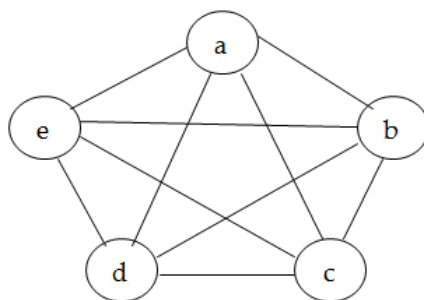
Таърифи математикӣ ва пурраи граф чунин аст: Системаи иборат аз се элемент $G=(X,U,F)$ -ро граф меноманд. Дар ин ҷо $X=X(x)$ -маҷмӯи қуллаҳо, $U=U(u)$ маҷмӯи тегаҳо ва F инъикоси байни онҳо мебошад.

Бигзор якчанд маҷмӯи охиринокм нуқтаҳо дода шуда бошад. Ин маҷмӯи додашударо қуллаҳои граф номида бо VG ишора мекунанд. Қуллаҳои графро одатан ба маҷмӯи ҳарфҳои зернависдор $VG = (V_1, V_2, \dots, V_n)$ ё ададҳо $VG = (1, 2, \dots, n)$ ишора мекунанд. Дар намуди геометрӣ бошад, ин маҷмуро бо дирачаҳо ишора мекунанд.

Дилхоҳ ҷуфти элементҳоро аз маҷмӯи VG тега (камон)-и граф номида бо EG ишорат мекунанд. Тег (камон)-ҳои графро одатан бо ҷуфти қуллаҳои граф ишора мекунанд:

$$EG = ((V_1, V_2), (V_3, V_4), \dots, (V_{n-1}, V_n)) \text{ ё } EG = ((1, 2), (3, 4), \dots, (n-1, n)).$$

Барои ба хонанда пурратар фаҳмо шудан, графери меорем, ки дорои 10 тега ва 5 қулла мебошад.

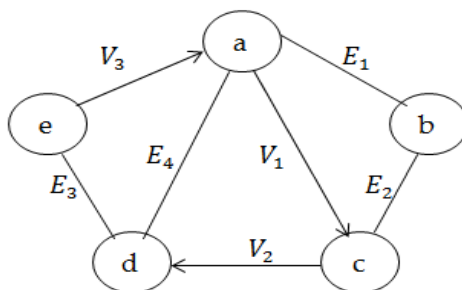


Дар баъзе адабиётҳо ҳангоми омӯзиши графҳо ба мафҳумҳои тега ва камон дучор шудан мумкин аст. Бояд қайд кард, ки барои графҳо мафҳуми тега аз камон фарқ мекунанд. Тег ин алокаи дутарафа ё роҳи дутарафа дар байни қуллаҳо буда, камон роҳи яктарафа мебошад. Бо суханҳои дигар тега ду қулларо бо ҳамдигар

пайваст мекунад ва камон бошад аз як қулла баромада ба қуллаи дигар ба охир мерасад.

Ба сифати мисоли роҳи дутарафа мардеро пеши назар меорем, ки он метавонад ба хонаи ҳамсоя ба меҳмонӣ рафта, сипас бо роҳи рафтааш ба хонаи худ бармегардад. Ба сифати камон бошад, парашютистеро меорем, ки он худро аз парашют ба замин меандозад. Пас аз оне, ки он ба замин мерасад дигар наметавонад ба ҷои худ баргардад.

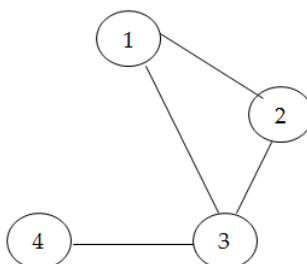
Одатан ҷуфти теғаҳоро бо ҳарфи E ва ҷуфти камонҳоро бо ҳарфи V ишора мекунанд. Ба сифати мисол дар поён граферо мерем, ки дорои панҷ қулла, чор теға ва се камон мебошад.



Дарачаи қуллаҳои графҳо

Дарачаи қуллаи граф гуфта, миқдори теғаҳои графро, ки ба ин қулла тааллуқ доранд меноманд. Дарачаи қулларо инчунин валентност низ меноманд ва бо $d(v)$ ё $\text{deg}(v)$ ишора мекунанд. Қуллаи графро, ки барои он $d(v) = 0$ аст қуллаи қайдкардашуда меноманд.

Қуллаи граф ҷуфт номида мешавад, агар барои он $d(v)$ адади ҷуфт бошад ва баракс.

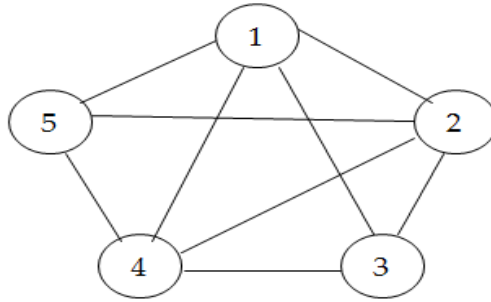


Дар ин ҷо $d(1)=2, d(2)=2, d(3)=3$ ва $d(4)=1$ мебошад.

Шакли геометрии графҳо

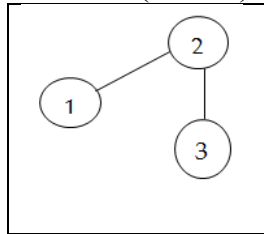
Ба намуди графикӣ пешниҳод намудани графҳо имконияти пурратар дарк намудани онҳоро фароҳам меорад. Графҳо на дар як шакл, балки бо шаклҳои гуногун дода мешаванд. Ҳоло баъзе аз шаклҳои асосии ба намуди графикӣ дода шудани графҳоро маври-ди баҳс қарор медиҳем.

1. Граф беаломатдор (неориентирный) номида мешавад, агар дар байни қуллаҳои он роҳи рафту омад вучуд дошта бошад. Навишти математикии чунин навъ графҳо шакли $G=(U,A)$ –ро дорад. Дар ин ҷо U -маҷмӯи қуллаҳо ва A -ро равиши теғаҳоро ифода мекунанд. Бо суханҳои дигар, графҳои беаломат гуфта, граферо меноманд, ки агар қуллаҳои онро танҳо теғаҳо пайваст кунанд. Дар поён тасвири графикаи ин навъ граф оварда шудааст.

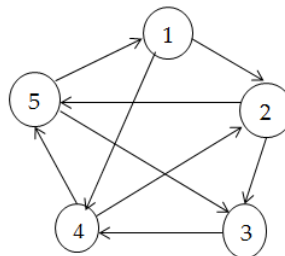


Ин граф дорои 5 қулла $VG=(1,2,3,4,5)$ ва 10 тега $EG=\{(1, 2), (1, 3), (1, 4), (1, 5), (2, 3), (2, 4), (2, 5), (3, 4), (3, 5), (4, 5)\}$ мебошад.

2. Графи беаломат содда номида мешавад, агар он дорои ҳалқа набуда, дилхоҳ чуфти қуллаҳои онро на зиёда аз як тег (камон) пайваст кунад.

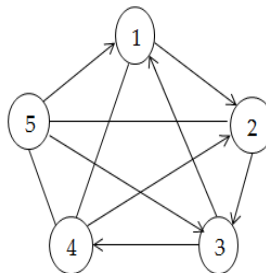


3. Граф равишдор (ориентирный) номида мешавад, агар дар байни қуллаҳои он роҳи яктарафа вуҷуд дошта бошад. Навишти математикии чунин навъ графҳо шакли $G=(U,A)$ –ро дорад. Дар ин ҷо U -ро маҷмӯи қуллаҳо ва A -ро равиши тегҳо меноманд. Бо суҳанҳои дигар, графҳои равишдор гуфта, граферо меноманд, ки қуллаҳои онро танҳо камонҳо пайваст мекунанд ё он танҳо аз камонҳо иборат бошад.



4. Граф омехта номида мешавад, агар дар байни баъзе қуллаҳои он равиш нишон дода шуда бошад дар баъзеашон не. Дар поён тасвири графҳои ин навъ граф оварда шудааст.

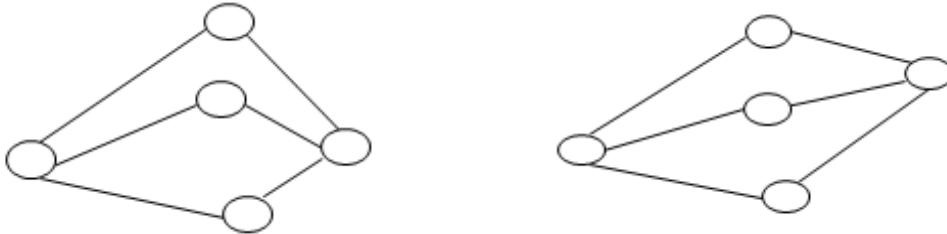
Дар ин ҷо тегҳои равишдор $E G_1 = \{(1, 2), (2, 3), (3, 1), (3, 4), (5, 1)\}$ буда, тегҳои беравиш $EG_1 = \{(2, 5), (4, 1), (4, 5), \dots\}$ мебошанд.



5. Графе, ки танҳо аз қуллаҳои қайдкардашуда иборат аст, сифрграф номида мешавад.

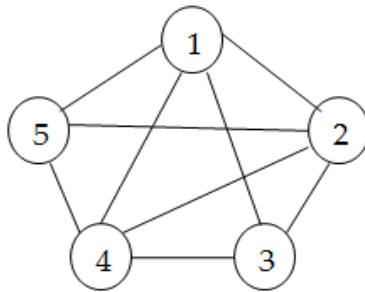


6. Ду гарф изоморфӣ номида мешаванд, агар миқдори тега ва қуллаҳои онҳо баробар буда, шаклҳои гуногун дошта бошанд.

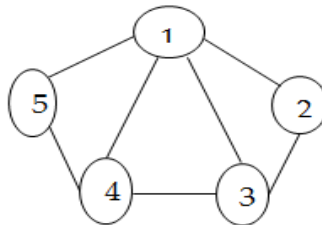


Тасдиқот. *Графҳои изоморфӣ бо ҳамдигар баробаранд.*

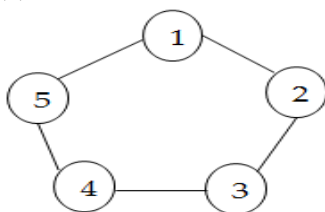
7. Граф пурра номида мешавад, агар аз дулхоқ қуллаи он имконияти гузаштан ба қуллаи дигар бе истифода аз қуллаи иловагӣ вуҷуд дошта бошад. Чунин навъ граф дорои $N*(N-1)/2$ тега мебошад. Дар ин ҷо N миқдори қуллаҳои графро ифода мекунад.



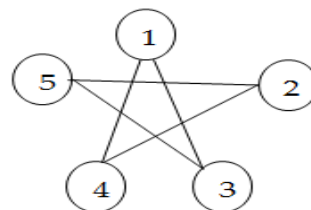
8. Граф нопурра номида мешавад, агар аққалан дар байни ду қуллаи он бе истифода аз қуллаи иловагӣ роҳ мавҷуд набошад.



9. Графи G_1 пуркунандаи графи G номида мешавад, агар графи G_1 ҳамаи қуллаҳои графи G -ро дошта, хангоми қуллаҳои онро ба графи G гузоштан, графи пурра ҳосил шавад.

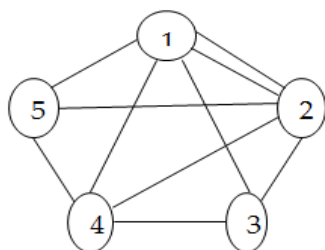


Графи G

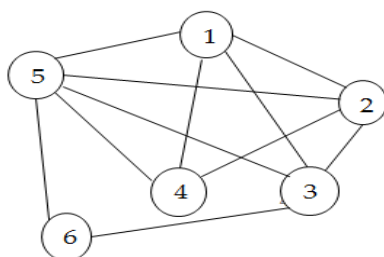


Графи G_1

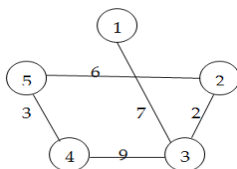
10. Граф қаратӣ номида мешавад, агар дар байни ду қуллаи он зиёда аз як роҳ вуҷуд дошта бошад.



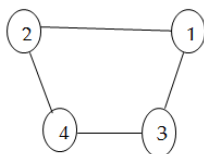
11. Граф Эйлерй номида мешавад, агар дараҷаи ҳамаи теғаҳои он чуфт бошанд.



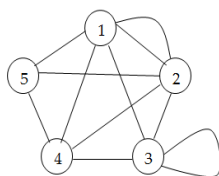
12. Граф вазнй номида мешавад, агар ба ҳар як теға (камон)-ҳои он адад мувофиқ гузошта шавад. Адади гузошташуда масофа, вазн ва ё дигар бузургиро ифода мекунад.



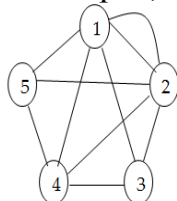
13. Граф якчинса номида мешавад, агар дараҷаи ҳамаи қуллаҳои он байни ҳам баробар бошанд. Бо суҳанҳои дигар агар миқдори элементҳои ғайринулии ҳамаи сатрҳои матритсаи ҳамсоғии он байни ҳам баробар бошанд.



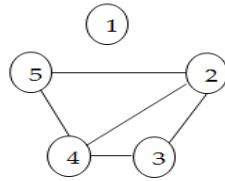
14. Псевдограф гуфта, граферо меноманд, ки дорои теғаҳои кратӣ ва ҳалқавӣ бошад.



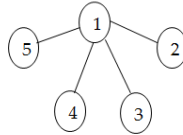
15. Пседографе ки дар он ҳалқа вучуд надорад, мултиграф номида мешавад.



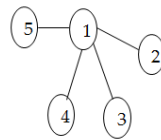
16. Агар дар байни ягон қуллаи граф роҳ мавҷуд набошад, чунин қулларо қуллаи қайдкардашуда меноманд.



17. Қуллае, ки дар он аз ҳамаи қуллаҳои дигар роҳ (теға ё камон) мавҷуд бошад, қуллаи ҳамсоғӣ номида мешавад.

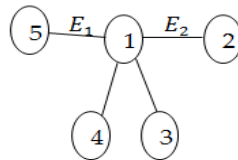


18. Қуллаҳое, ки онҳоро теға ё камон бо ҳам пайваст мекунад, ҳамсоя номида мешавад.



Дар ин ҷо қуллаҳои 5 ва 1 ҳамсоя мебошанд.

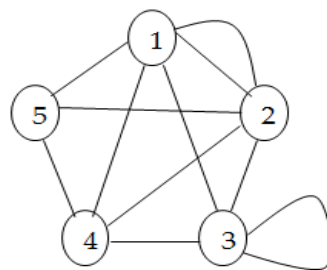
19. Ду теға ё камоне, ки дорои як қуллаи умумӣ мебошанд, ҳамсоя номида мешаванд.



Дар ин ҷо теғаҳои E_1 ва E_2 ҳамсоя мебошанд.

20. Агар қуллаҳои V_1, V_2, \dots, V_n бо қуллаи V_0 ҳамсоя бошанд, онгоҳ онҳоро аторофи қуллаи V_0 меноманд.

21. Қуллае, ки аз он роҳ оғоз гардида бо он ба охир мерасад ҳалқа (петлей) номида мешавад.



Дар ин ҷо қуллаи 3 ҳалқа (петлей) мебошад.

22. Дар графӣ равишдор роҳ гуфта, пайдарпайии теғаҳои равишдори онро меноманд.

23. Роҳи Эйлери гуфта, роҳеро меноманд, ки аз ҳамаи теғаҳои граф мегузарад.

24. Занҷири Эйлери гуфта, роҳеро меноманд, ки аз ҳамаи теғаҳои граф якбор мегузарад.

25. Агар қуллаи аввал ва охир дар занҷир ба ҳамдигар монанд бошанд, онгоҳ занҷирро ҳалқа меноманд.

26. Ҳалқа содда номида мешавад, агар ҳамаи қуллаҳои он ба ғайр аз қуллаи ибтидоӣ ва интиҳоӣ гуногун бошанд.

Намоиши дискретии графҳо

Тасвири геометрии графҳо яке аз усулҳои маъмули татқиқи графҳо ба қисоб меравад, вале ин усулро наметавон дар мошинҳои ҳисобкунии истифода кард. Аз ин ҷо зарурати истифодаи усули дискретӣ пеш меояд. Тавассути ин усул метавон ҳалли адабии графро дар МЭҲ ба даст овард. Барои иҷрои ин амал усулҳои гуногуну мавҷуд аст, ки асоситаринашон инҳо мебошанд:

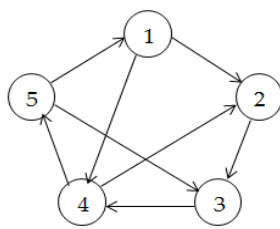
- 1) Матритсаи ҳамсоғӣ;
- 2) Руйхати ҳамсоғӣ;
- 3) Руйхати қуллаҳо;
- 4) Матритсаи якзайлӣ;
- 5) Матритсаи вазнӣ;
- 6) Руйхати теғаҳо;

Матритсаи ҳамсоғӣ

Матритсаи ҳамсоғӣ аз матритсаҳои дигар фарқи қулли дорад, зеро элементҳои он танҳо аз 0 ва 1 иборат мебошанд. Ин навъ матритсаҳо аз тасвири геометрии граф бо роҳи зерин ҳосил карда мешаванд: агар дар байни ду қулла роҳ мавҷуд бошад, онгоҳ элементи ба он мувофиқи матритса дорои қимати 1 ва дар акси ҳол дорои қимати 0 мешавад. Бо суҳанҳои дигар:

$$a_{ij} = \begin{cases} 1, & \text{агар аз } i \text{ ба самти } j \text{ роҳ мавҷуд бошад,} \\ 0, & \text{дар дигар ҳолатҳо.} \end{cases}$$

Барои пурратар ба хонанда фаҳмо шудан мисолеро дида мебароем. Бигзор графи равишдории зерин дода шуда бошад.



Матритсаи ҳамсоғии ин граф шакли зеринро дорад:

	1	2	3	4	5
1	0	1	0	1	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	1	0	0	1
5	1	0	1	0	0

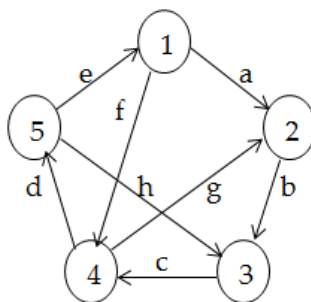
Руйхати ҳамсоғӣ

Барои сохтани чунин руйхат чадвали дорои ду сутун тасвир намуда, дар сарлавҳаи сутуни якум қуллаҳои баромад ва дар сарлавҳаи сутуни дуввум қуллаҳои даромад менависем. Дар сутуни якуми чадвали мазкур руйхати қуллаҳоро навишта, дар сутуни дуввум ҳамаи роҳҳои баромад аз қуллаи чориро менависем. Руйхати ҳамсоғӣ барои графи қабли бо шакли зерин сохта мешавад:

Куллаи баромад	Куллаи даромад
1	2, 4
2	3
3	4
4	2, 5
5	1, 3

Руйхати қуллаҳо

Руйхати қуллаҳо аз ҷуфти қуллаҳои ҳамсоғӣ сохта мешаванд. Барои иҷрои ин амал бояд, қуллаҳоро бо ягон ҳарф ишора карда сипас ҷадвали дорои се сатр месозем, ки дар сатри якуми ҷадвал қуллаҳои баромад, дар сатри дуввуми ҷадвал қуллаҳои даромад ва дар сатри сеюм дарозии теғҳо (ишораи кардаамон) навишта мешавад. Бигзор графи равишдори зерин дода шуда бошад.



Руйхати қуллаҳои ин граф чунин сохта мешаванд.

1	2	3	4	5	1	4	5
2	3	4	5	1	4	2	3
a	b	c	d	e	f	g	h

Матритсаи якзайлӣ

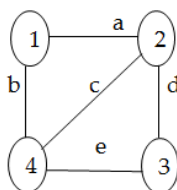
Баръакси матритсаҳои ҳамсоғӣ, ки квадрати буданд, матритсаи якзайли дорои n сатр ва m мебошад. Дар ин n миқдори қуллаҳои граф ва m миқдори теғҳои графро ифода мекунад. Матритсаи якзайли аз рӯи қулла ба теғ сохта мешавад, на аз рӯи теғ ба теғ.

Тарзи соختани матритсаи якзайлӣ барои графҳои равишдор ва беравиш фарқ мекунад.

Тарзи пур кардани матритсаи якзайли барои графи беравиш чунин аст:

$$a_{ij} = \begin{cases} 1, & \text{агар қуллаи } V_i \text{ бо теғи } E_j \text{ якзайл бошад,} \\ 0, & \text{дар дигар ҳолатҳо.} \end{cases}$$

Бигзор графи беравиши зерин дода шуда бошад.



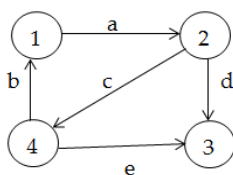
Матритсаи якзайлии онро месозем.

	a	b	c	d	e
1	1	1	0	0	0
2	1	0	1	1	0
3	0	0	0	1	1
4	0	1	1	0	1

Матритсаи якзайли барои графҳои равишдор ба сурати зерин пур карда мешавад:

$$a_{ij} = \begin{cases} -1, & \text{агар қуллаи } V_i \text{ ибтидои теғайи } E_j \text{ бошад,} \\ +1, & \text{агар қуллаи } V_i \text{ интиҳои теғайи } E_j \text{ бошад,} \\ 0, & \text{агар қуллаи } V_i \text{ бо теғайи } E_j \text{ якзайл набошад.} \end{cases}$$

Бигзор графҳои равишдори зерин дода шуда бошад.

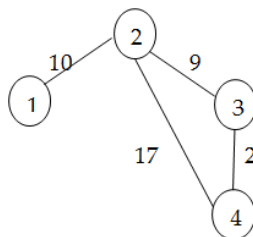


Матритсаи якзайлии онро месозем.

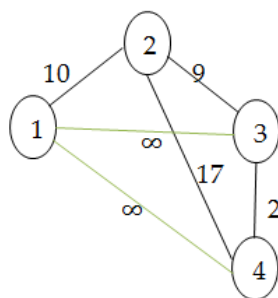
	a	b	c	d	e
1	1	-1	0	0	0
2	-1	0	1	1	0
3	0	0	0	-1	-1
4	0	1	-1	0	1

Матритсаи вазнӣ

Барои графҳои вазнӣ низ метавон матритса сохт. Матритсаи графҳои вазнӣ шакли $w = (x_{ij})$ -ро дорад, ки дар ин ҷо x_{ij} вазни теғайе, ки аз қуллаҳои i ва j -ро пайваст мекунад мебошад. Агар дар теғайи вазн нишон дода нашуда бошад, онгоҳ ба сифати вазн вобаста аз шартҳои масъала $+\infty$ ё 0 навишта мешавад. Агар граф пурра набошад, мувофиқи теорема аз риёзиёти дискретӣ (Дилхоҳ графҳои нопурраро метавон ба шакли пурра овард) он ба шакли пурра оварда мешавад. Ба сифати мисол графҳои зеринро дида мебароем:



Аз сабаби он ки граф пурра нест, бинобар ин аввал онро ба намуди пурра меорем. Намуди пурраи он шакли зеринро дорад:



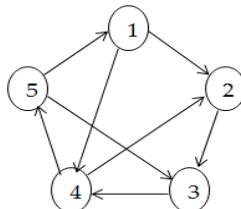
Пас аз оне, ки граф ба шакли пурра оварда шуд. Матритсаи вазнии онро месозем.

∞	10	∞	∞
10	∞	9	17
∞	9	∞	2
∞	17	2	∞

Барои ёфтани роҳи кутохтарин дар чунин графҳо аз алгоритми Прима истифода мебарем.

Руйхати теғаҳо

Барои графҳои равишдор усули дигари намоиш вучуд дорад. Дар ин усул ду массив истифода бурда мешавад, ки дар массиви якум ибтидои теғаҳо ва дар массиви дуввум интиҳои теғаҳо навишта мешаванд. Андозаи массивҳо ба миқдори теғаҳо баробар мебошанд. Ба сифати мисол, графҳои зеринро дида мебароем.



$r=(1, 1, 2, 3, 4, 4, 5, 5);$

$t=(2, 4, 3, 4, 2, 5, 1, 3);$

Дар ин ҷо r баромад (ибтидо)-и теғаҳо ва t даромад (интиҳо)-и теғаҳоро ифода мекунад.

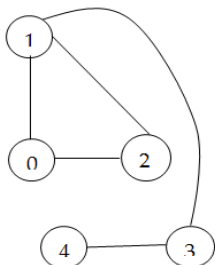
Чустучӯ дар умқ (Depth-first search)

Яке аз усулҳои маъмули систематикӣ татқиқи ҳамаи қуллаҳои граф-методи чустучӯ дар умқ ба ҳисоб мебошад. Барои графҳои беравиши $G=(X,U,F)$ чустучӯ дар умқ ба сурати зерин амали карда мешавад:

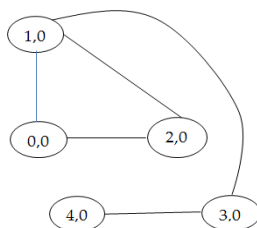
Агар дар қуллаи $x \in X$ қарор дошта бошем ба яке аз теғаҳои $(x,y) - e$, ки ба қуллаи $y \in X$ якзайлӣ мебошанд, ҳаракат мекунем. Агар қуллаи y аллакай аз назар гузаронида шуда бошад, ба x баргашта теғаи дигареро интиҳоб мекунем. Агар қуллаи y аз назар гузаронида нашуда бошад, ба он ҳаракат намуда ҷарайни рекурсивиро барои он татбиқ мекунем. Агар ҳамаи теғаҳои ба қуллаи x якзайлӣ аз назар гузаронида шуда бошанд, қафо ба теғаи (s,x) , ки аз онҷо ба x омада будем баргашта, амали татқиқро барои теғаҳои якзайлӣ қуллаи $s \in X$ давом медиҳем. Ин амал ҳамон вақт ба анҷом мерасад, агар ҳамаи теғаҳоро аз назаргузаронида дар қуллаи ибтидоӣ баргардем.

Чустучӯ дар умкӯро инчунин метавон барои графҳои равишдор низ анҷом дод. Бигзор дар гиреҳи x -и графи равишдор қарор дошта бошем. Теғайи (x, y) (теғайе, ки аз x мебарояд) ро интиҳоб карда, ҳамаи теғайҳои аз y баромадаро татқиқ карда қафо ба теғайи x бармегардем. Ва онҷо теғайи дигарро интиҳоб карда ин амалро барои он низ анҷом медиҳам. Ба ҳамин монанд теғайҳои граф татқиқ карда мешаванд.

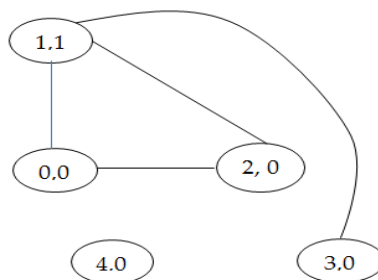
Барои ба хонанда пурратар фаҳмо шудан мисолеро дида мебароем. Бигзор графи беравиши зерин дода шуда бошад.



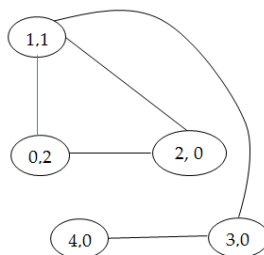
Дар қадами якум ба ҳамаи қуллаҳои граф нишонаи 0 мегузорем.



Ба сифати қуллай ибтидоӣ қуллай 1-ро интиҳоб мекунем ва ба он нишонаи 1 мегузорем. Ин нишондиҳандаи он аст, ки татқиқ аз онҷо оғоз гардидааст.

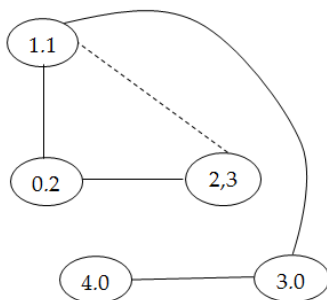


Дар қадами сеюм теғайе, ки ба қуллай 1 пайваст буда, дорои рақами тартибии хурдтарин мебошад интиҳоб мекунем. Дар ҳолати мо ин қуллай 0-ум мебошад. Ба қуллай мазкур нишонаи 2 мегузорем.

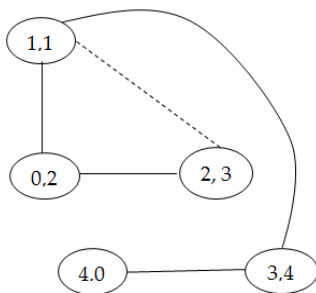


Акнун амали чустучӯро аз қуллай 0 оғоз мекунем. Барои ин теғайе, ки қуллай 0-умро бо қуллай нишонаназуштаи рақами тартибиаш хурдтарин пайваст мекунад интиҳоб мекунем. Дар ҳолати мо ин қуллай 2 мебошад. Ба қуллай интиҳобкардамон нишонаи 3 мегузорем. Пас аз ин амал теғайи $(0, 2)$ аз назаргузаронидашуда ба ҳисоб меравад.

Амали ҷустуҷӯро аз қуллаи 2 оғоз мекунем. Аз он танҳо як теға (2,1) мебарояд. Аз дигар тараф қуллаи 1 аллакай дорои нишона мебошад, бинобар ин зарур аст, ки теғаи (2,1) баракс гузошта шавад. Чунин теға бо тире-тире ишора карда мешавад.



Ба қуллае, ки аз онҳо ба қуллаи 2 омада будем бармегардем. Ин қулла 0 мебошад. Аз ин қулла ягон теғаи бенишона намебарояд. Аз ин ҷо ба қуллаи 1 бармегардем. Амали татқиқро аз ин қулла оғоз мекунем. Аз онҳо танҳо як теғаи бенишона (1, 3) мебарояд. Аз ин ҷо қуллаи 3 дорои нишонаи 4 мегардад.



Амали ҷустуҷӯро аз қуллаи 3 оғоз мекунем. Аз ин қулла танҳо як теғаи бенишона (3, 4) мебарояд. Аз ин ҷо барои қуллаи 4 нишонаи 5 мегузorem.



Аз сабаби он, ки аз қуллаи 4 ягон теғаи бенишона намебарояд, бинобар ин, ба қуллаи 3 бармегардем. Аз қуллаи 3 низ ягон теғаи бенишона намебарояд. Аз онҳо ба қуллаи 1 ки амали ҷустуҷӯро аз он оғоз карда будем, яъне қуллаи 1 бармегардем. Ва ба ҳамин алгоритм корашро ба анҷом мерасонад.

Татбиқи алгоритми ҷустуҷӯ дар умқ

Барои татқиқи алгоритми ҷустуҷӯ дар умқ аввал матритсаи ҳамсоғии графро, ки шакли зеринро дорад менависем.

	0	1	2	3	4
0	0	1	1	0	0
1	1	0	1	1	0
2	1	1	0	0	0
3	0	1	0	0	1
4	0	0	0	1	0

Барои татбиқи методи ҷустуҷӯ дар умқи як массив барои нишона ва як пушта истифода бурда мешавад. Дар қадами нахуст массиви нишона ва пуштаро эҷод мекунем.

Массиви нишона

Рақами тартибӣ	0	1	2	3	4
Элемент	0	0	0	0	0

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	-	-	-	-	-

Қаблан дар ҳолати графикаи мо барои татқи аз қуллаи 1 оғоз карда будем, бинобар ин, рақами тартибии онро ба пушта гузошта барои он нишонаи 1 мегузorem.

Массиви нишона

Рақами тартибӣ	0	1	2	3	4
Элемент	0	1	0	0	0

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	1	-	-	-	-

Дар қадами сеюм тегаеро, ки қуллаи якро пайваст намуда, дорои рақами тартибии хурдтарин мебошад ва то ҳол дорои нишона намебошад интихоб мекунем. Дар ҳолати мо ин қуллаи қуллаи 0-ум мебошад. Қуллаи 0-ро ба пушта гузошта, ба он нишонаи ду мегузorem.

Массиви нишона

Рақами тартибӣ	0	1	2	3	4
Элемент	2	1	0	0	0

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	0	1	-	-	-

Дар ин ҳолат тегаи (1,0) аз назаргузаронидашудаи роста ба ҳисоб рафта, дар бурриши сатри якум ва сутуни 0-уми матритсаи ҳамсоғӣ рақами якро ба ду иваз мекунем.

Амали ҷустуҷӯро аз қуллаи 0-ум оғоз мекунем. Сатри сифруми матритсаи ҳамсоғиро аз назар мегузaronем. Дар ин сатр элементи якуми ба 1 баробарро пайдо мекунем. Дар ҳолати мо ин элемент дар сутуни дуюм мехобад. Аз ин ҷо қуллаи дуввумро ба пушта гузошта, ба он нишонаи 3 мегузorem.

Массиви нишона

Рақами тартибӣ	0	1	2	3	4
Элемент	2	1	3	0	0

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	2	0	1	-	-

Пас аз ин сатри сеюми (рақами 2) матритсаро аз назар мегузaronем. Дар ин ҷо элементи $\text{Graf}[2][1]=1$ мебошад. Ин нишондиҳандаи тегаест, ки қуллаи 1 ва 2-ро пайваст мекунад, вале аз сабаби он, ки қуллаи 1 аллакай аз назар гузаронидашуда мебошад, бинобар ин зарур аст, ки қуллаи 2 аз пушта нест карда шавад. Пас аз ин ишорагари қуллаҳо як катакча ба тарафи чап кучонида мешавад.

Рақами тартибӣ	0	1	2	3	4
Элемент	0	1	-	-	-

Дар ин ҳолат теғай (0, 2) баракс ҳисобида мешавад. Агар теғай қуллай i ва j -ро бо ҳам пайваस्तкунанда баракс бошад, онгоҳ элементи мувофиқи сатри i ва сутуни j -уми матритсаи ҳамсоғӣ ба 3 иваз карда мешавад.

Ба ҳамин монанд амали ҷустуҷӯро аз қуллай 0-ум оғоз мекунем. Аз сабаби он, ки аз қуллай 0-ум ягон теғай бенишона намебарояд, ба қафо ба қуллай 1-ум бармегардем. Дар ин ҳолат низ аз пушта як элементи дигарро низ нест мекунем. Дар натиҷаи ин амал пушта шакли зеринро мегирад.

Рақами тартибӣ	0	1	2	3	4
Элемент	1	-	-	-	-

Сатри якуми матритсаи ҳамсоғиро аз назар мегузаронем. Дар ин сатр элементи сутуни шашум ба 1 баробар мебошад. Бинобар ин, қуллай 3 ба пушта гузошта шуда, дорои нишонаи 4 мегардад. Пас аз ин амал граф ва пуштамон шакли зеринро мегирад.

Массиви нишона

Рақами тартибӣ	0	1	2	3	4
Элемент	2	1	3	4	0

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	4	1	-	-	-

Акнун сатри сеюми матритсаи ҳамсоғиро аз назар мегузаронем. Дар ин сатр элементи сутуни 4-ум ба як баробар мебошад. Аз ин ҷо қуллай 4 ба пушта гузошта шуда, қуллай 4-ум дорои нишонаи 5 мегардад.

Массиви нишона

Рақами тартибӣ	0	1	2	3	4
Элемент	2	1	3	4	5

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	5	4	1	-	-

Акнун қуллай чорумро аз назар мегузаронем, аз онҷо ягон теғай бенишона намебарояд, бинобар ин аз он аз пушта нест карда мешавад.

Пушта

Рақами тартибӣ	0	1	2	3	4
Элемент	4	1	-	-	-

Аз қуллаҳои 4 ва як низ ягон теғай бенишона намебарояд, бинобар ин онҳо низ аз пушта нест карда мешаванд.

Пас аз иҷрои ин амалҳо пушта ҳолӣ карда мешавад. Ба ҳамин монанд мо ба қуллай ибтидоӣ бармегардем. Дар охир матритсаи ҳамсоғӣ шакли зеринро мегирад:

	0	1	2	3	4
0	0	2	2	0	0
1	2	3	1	1	0
2	2	2	0	0	0
3	0	2	0	0	2
4	0	0	0	2	0

Дар охир ба намуди чамъбасти алгоритми ҳосил намудани матритсаи ҳамсоғии натиҷавиро меорем, ки шакли зеринро дорад:

$$Graf_{i,j} = \begin{cases} 2, & \text{агар теғай аз назаргузаронидашуда роста бошад,} \\ 3, & \text{агар теғай аз назаргузаронида чаппа бошад.} \end{cases}$$

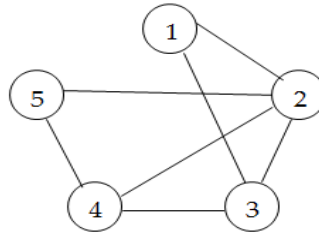
Пас аз шинос шудан ба моҳияти алгоритм дар мисоле татбиқи онро меорем.

```
#include <iostream>
using namespace std;
const int n=5;
int i, j;
bool *visited=new bool[n];
//матритсаи ҳамсоғии граф
int graph[n][n] =
{
    {0, 1, 1, 0, 0},
    {1, 0, 1, 1, 0},
    {1, 1, 0, 0, 0},
    {0, 1, 0, 0, 1},
    {0, 0, 0, 1, 0}
};

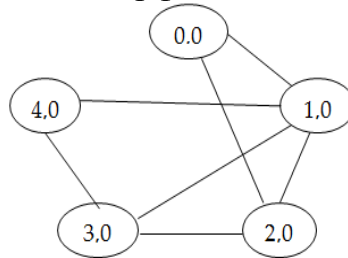
void DFS(int st)    {
    int r;
    cout<<st+1<<" ";
    visited[st]=true;
    for (r=0; r<n; r++)
        if ((graph[st][r]!=0) && (!visited[r]))
            DFS(r);
}
//Функсияи асосӣ
int main()
{
    int start;
    cout<<"матритсаи ҳамсоғии граф: "<<endl;
    for (i=0; i<n; i++)
    {
        visited[i]=false;
        for (j=0; j<n; j++)
            cout<<" "<<graph[i][j];
        cout<<endl;
    }
    cout<<"теғай ибтидоӣ" >> " "; cin>>start;
    // Массиви теғаҳое, ки тавассути онҳо ҳаракат карда шудааст..
    bool *vis=new bool[n];
    cout<<"тартиби ҷустуҷӯ: ";
    DFS(start-1);
    delete []visited;
    system ("pause");
    return 0;
}
```

Љустуҷӯ дар васеъгӣ

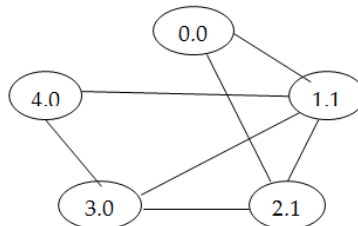
Љустуҷӯ дар васеъгӣ яке аз методҳои маъмули татқиқи теға ва қуллаҳои граф ба ҳисоб меравад. Методи мазкурро бо мисоле дида мебароем. Бигзор графи беравиши зерин дода шуда бошад.



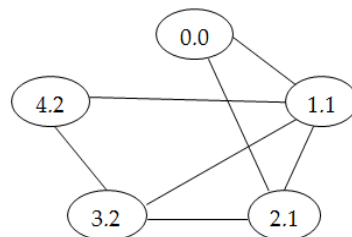
Пеш аз оғози татқиқи граф ба ҳар як қуллаи он ягон нишона мегузорем. Бигзор ин нишона 0 бошад. Пас аз ин графамон шакли зеринро мегирад:



Пас аз ин қуллаи ибтидоӣ (қуллае, ки бояд татқиқро аз он оғоз кунем)-ро интихоб мекунем. Барои намуна қуллаи нулумро интихоб мекунем. Пас аз интихоби қуллаи ибтидоӣ барои он ва барои ҳамаи қуллаҳои аз бароянда нишонаи 1 мегузорем. Дар натиҷа графамон шакли зеринро мегирад:



Пас аз ин амал қуллаҳои 1 ва 2 дорой нишонаи 1 мегарданд. Акнун падарпай қуллаҳои дорой нишонаи 1-ро аз назар гузаронида барои қуллаҳои аз онҳо бароянда нишонаи 2 мегузорем. Пас аз анҷоми ин амал графамон шакли зеринро мегирад.



Пас аз иҷрои ин амал қуллаҳои 3 ва 4 дорой нишонаи 2 мешаванд. Баъд аз ин қуллаҳои аз 3 ва 4 баромадаро татбиқ мекунем. Аз сабаби оне, ки ҳамаи қуллаҳо аллақай аз назар гузаронида шудаанд, алгоритм қорашро ба охир мерасонад.

Пас аз анҷоми кори алгоритм ҳар як қулла дорой нишонаи худ мегардад.

Амалисозии ин алгоритм дар барнома ба сурати зерин аст:

1) Матритсаи ҳамсоягии графӣ додашударо месозем, ки он шакли зеринро дорад:

	0	1	2	3	4
0	0	1	1	0	0
1	1	0	1	1	1
2	1	1	0	1	0
3	0	1	1	0	1
4	0	1	0	1	0

2) Барои нигоҳ доштани нишокаи қуллаҳои граф вобаста ба миқдори қуллаҳо як массив муаррифӣ мекунем. Тавре ки аз расмҳо дидам дар қадами нахустин ба ҳамаи қуллаҳои граф нишокаи 0 бахшида шуда буд. Ба ҳамин монанд ҳангоми амалисозии алгоритм дар қадами нахустин ба сифати қимати ибтидоӣ барои элементҳои массив ягон адади ба қадри кифоя калонро мебахшем.

Массиви нишокаи

Рақами тартибӣ	0	1	2	3	4
Элемент	32767	32767	32767	32767	32767

Илова бар ин як саф низ муаррифӣ мекунем, ки нишокаи қуллаи ҷориро дар худ нигоҳ дорад. Дар ибтидо сафро ҳолӣ мегузорем.

Рақами тартибӣ	0	1	2	3	4
Элемент	-	-	-	-	-

Пас аз ин амал зарур аст, ки қуллаи ибтидоиро барои татқиқ интихоб кунем. Ба сифати қуллаи ибтидоӣ метавон дилхоҳ қулларо интихоб кард. Барои намуна қуллаи нолумро интихоб мекунем. Пас аз интихоб намудани қуллаи ибтидоӣ, ба он нишокаи 0 гузошта, рақами тартибии онро ба саф мегузорем.

Массиви нишокаи

Рақами тартибӣ	0	1	2	3	4
Элемент	0	32767	32767	32767	32767

саф

Рақами тартибӣ	0	1	2	3	4
Элемент	0	-	-	-	-

Акнун сатри сифруми матритсаи ҳамсоягиро аз назар мегузоро-нем. Агар дар ин сатр ягон элементҳои ғайринулӣ пайдо шавад, онгоҳ рақами тартибии он ба саф гузошта мешавад. Барои ҳолати мо ин қуллаҳои 1 ва 2 ба саф гузошта шуда, дорои нишокаи 1 мегарданд. Аз ин ҷо массиви нишокаи ва саф шакли зеринро мегиранд.

саф

Рақами тартибӣ	0	1	2	3	4
Элемент	0	1	2	3	4

Массиви нишокаи

Рақами тартибӣ	0	1	2	3	4
----------------	---	---	---	---	---

Элемент	0	1	1	32767	32767
---------	---	---	---	-------	-------

Пас аз ин амал элементи 0 аз саф нест карда шуда, ишорагари элементи якум як катакча ба пеш кӯчонида мешавад. Акнун атрофи қуллаҳои 1 ва 2-ро аз назар мегузаронем. Агар ягон элементи сатри якуми матритсаи ҳамсоягӣ ба нул баробар набошад, онгоҳ рақами тартибӣ он ба саф гузошта мешавад. Дар ҳолати мо қуллаҳои 2, 3 ва 4 ба саф гузошта шуда, дорои нишонаи 2 мегарданд.

саф					
Рақами тартибӣ	0	1	2	3	4
Элемент	0	1	2	3	4

Массиви нишона					
Рақами тартибӣ	0	1	2	3	4
Элемент	0	1	1	2	2

Пас аз ин амал элементи 1 аз саф нест карда шуда, ишорагари саф як хона ба тарафи пеш кӯчонида мешавад.

Акнун қуллаи дуҷуми матритсаро аз назар мегузаронем. Аз сабаби он, ки дар атрофи он ягон қуллаи бе нишона вучуд надорад, элементи 2 аз саф нест карда мешавад ва ишорагари элементи якум як катакча ба тарафи рост кӯчонида мешавад. Ба ҳамин монанд атрофи қуллаҳои 3 ва 4 низ аз назар гузаронида мешаванд. Аз сабаби он, ки дар атрофи ин қуллаҳо қуллаи бенишона вучуд надорад, пайдарпай ин қуллаҳо аз саф нест карда мешаванд.

Пас аз назаргузарониҳои ҳамаи тегаҳо саф ҳолӣ гардида кори алгоритм ба охир мерасад.

Барои ба хонанда пурратар фаҳмо шудан барномаи ин равиш дар поён оварда шуддааст.

```
#include <iostream>
using namespace std;
const int n=5;
int i, j;
//матритсаи ҳамсоягии граф
int GM[n][n] =
{
    {0, 1, 1, 0, 0},
    {1, 0, 1, 1, 1},
    {1, 1, 0, 1, 0},
    {0, 1, 1, 0, 1},
    {0, 1, 0, 1, 0},
};

void BFS(bool *passed, int unit)
{
    //навбат
    int *queue=new int[n];
    //ишорагари навбат
    int count, head;
    for (i=0; i<n; i++) queue[i]=0;
    count=0; head=0;
    queue[count++]=unit;
```

```

    passed[unit]=true;
    while (head<count) {
        unit=queue[head++];
        cout<<unit+1<<" ";
        for (i=0; i<n; i++)
            if (GM[unit][i] && !passed[i]) {
                queue[count++]=i;
                passed[i]=true;
            }
    }
    delete []queue;
}
//Функсияи асосӣ
int main(){
    int start;
    cout<<"Қуллаи ибтидоиро дохил кунед >> "; cin>>start;
    // Массиви теғаҳое, ки тавассути онҳо ҳаракат карда шудааст..
    bool *vis=new bool[n];
    cout<<"матритсаи ҳамсоягии граф: "<<endl;
    for (i=0; i<n; i++)
    {
        vis[i]=false;
        for (j=0; j<n; j++)
            cout<<" "<<GM[i][j];
        cout<<endl;
    }
    cout<<" тартиби ҷустуҷӯ: ";
    BFS(vis, start-1);
    delete []vis;
    return 0;
}

```

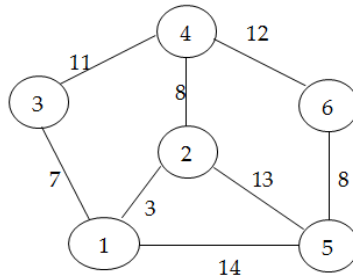
Сохтони роҳи кутоҳтарин

Масъалаҳои сохтани роҳи кутоҳтарин, яке аз масъалаҳои муҳиме ба ҳисоб меравад. Масалан бигзор N шаҳр мавҷуд аст. Барои ҳар ҷуфти шаҳрҳо (I, J) метавон роҳ сохт, ки ин шаҳрҳоро бо ҳамдигар бе гузаштан аз шаҳри дигар пайваст кунад. Нархи ҷунин роҳ ба $A(I, J)$ баробар мебошад.

Барномае нависед, ки аз ҳама роҳи арзонтаринро муайян кунад. Ин алгоритм иҷозати аз дилхоҳ шаҳр гузаштан ба шаҳри дигарро бе истифода аз шаҳри дигар фароҳам орад. Барои ҳалли ҷунин масъала, зарур аст, ки аз графҳои вазндор истифода бурда шавад.

Барои ёфтани роҳи кутоҳтарин аз алгоритми Дейскри истифода мекунем.

Барои пурратар ба моҳияти ин масъала шинос шудан, аввал ин шаҳрҳоро ба сурати граф тасвир мекунем.



Бигзор зарурати муайян намудани роҳи кутуҳтарини байни қуллаҳои 1 ва 4 пеш ояд. Барои ин аввал матритсаи вазнии (хамсоягии) графро месозем, ки шакли зеринро дорад:

	1	2	3	4	5
1	0	3	7	0	14
2	3	0	0	8	13
3	11	0	0	11	0
4	0	8	11	0	0
5	14	13	0	0	0

Ин масъаларо бо истифода аз алгоритми Дейскри ҳал мекунем. Барои ин алгоритм аз ду массив бо номҳои Label ва Active истифода мекунем. Ба сифати элемент дар массиви Label[i] масофа аз қуллаи ибтидоӣ то қуллаи i-юмро нигоҳ медорем. Дар массиви Active бошад, нишонаи қуллаҳои аз назаргузаронидашуда нигоҳ дошта мешаванд. Аз сабаби онки 1 қуллаи ибтидоӣ мебошад, пас сифати элементи яқум дар массиви Active нишонаи 1 мегузорем.

Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	1	0	0	0	0	0

Массиви Label

Рақами тартибӣ	1	2	3	4	5	6
Элемент	0	32767	32767	32767	32767	32767

Аз қуллаи 1 се теға мекунем: (1, 2), (1, 3) ва (1, 5). Агар барои ягонто аз қуллаҳои ёфташуда шарт $Label[j] > Label[i] + Graf[i][j]$ иҷро гардад, пас ин қулла аз назаргузаронидашуда ба ҳисоб рафта, элементҳои массиваи нишона чунин пур карда мешаванд:

$$Label[j] = Label[i] + Graf[i][j];$$

Барои ҳолати мо:

$$1) Label[2] > Label[1] + Graf[1][2] \Rightarrow 32767 > 1 + 3. \text{ Пас аз ин чо}$$

$$Label[2] = Label[1] + Graf[1][2] = 0 + 3 = 3.$$

$$2) Label[3] > Label[1] + Graf[1][3] \Rightarrow 32767 > 1 + 7. \text{ Пас аз ин чо}$$

$$Label[3] = Label[1] + Graf[1][3] = 1 + 7 = 8.$$

$$3) Label[5] > Label[1] + Graf[1][5] \Rightarrow 32767 > 1 + 14. \text{ Пас аз ин чо}$$

$$Label[5] = Label[1] + Graf[1][5] = 1 + 14 = 15.$$

Пас аз ин амал массивҳои мо шакли зеринро мегиранд:

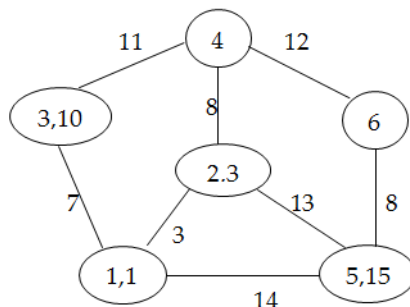
Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	1	1	1	0	1	0

Массиви Label

Рақами тартибӣ	1	2	3	4	5	6
Элемент	1	3	8	32767	15	32767

Пас аз ин амал графамон шакли зеринро мегирад.



Амали ҷустуҷӯ аз қуллае, ки нишонаи хурдтарин дорад, оғоз карда мешавад. Ин қуллаи 2-юм мебошад. Аз ин қуллаи се теға (2, 1), (2, 4), (2, 5) мебарояд. Барои ин қуллаҳо низ шарти

$Label[j] > Label[i] + Graf[i][j]$ -ро месанҷем.

1) $Label[1] > Label[2] + Graf[2][1] \Rightarrow 1 < 4 + 3$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 1 бетағйир гузошта мешавад.

2) $Label[4] > Label[2] + Graf[2][4] \Rightarrow 232767 > 4 + 8$. Пас аз ин ҷо $Label[4] = Label[2] + Graf[2][4] = 3 + 8 = 11$.

3) $Label[5] > Label[2] + Graf[2][5] \Rightarrow 15 > 4 + 13$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 5 бетағйир гузошта мешавад.

Пас аз ин амал массивҳои мо шакли зеринро мегиранд:

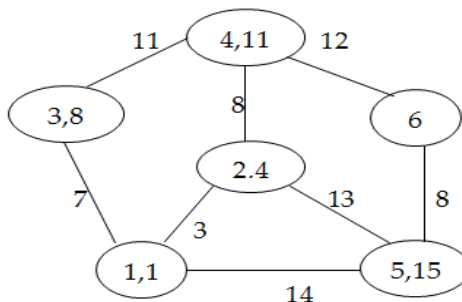
Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	1	0	1	1	1	0

Массиви Label

Рақами тартибӣ	1	2	3	4	5	6
Элемент	1	4	8	11	15	32767

Пас аз ин амал графамон шакли зеринро мегирад.



Амали ҷустуҷӯро аз қуллаи 1 оғоз мекунем. Аз ин қулла се теға (1, 2), (1, 3) ва (1, 5) мебарояд. Агар барои ягонто аз қуллаҳои ёфташуда шарти $Label[j] > Label[i] + Graf[i][j]$ иҷро гардад, пас ин қулла аз назаргузаронидашуда ба ҳисоб рафта, элементҳои массиви нишона чунин пур карда мешаванд:

$Label[j] = Label[i] + Graf[i][j]$;

Барои ҳолати мо:

1) $\text{Label}[2] > \text{Label}[1] + \text{Graf}[1][2] \Rightarrow 4 > 1 + 3$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 2 бетағйир гузошта мешавад.

2) $\text{Label}[3] > \text{Label}[1] + \text{Graf}[1][3] \Rightarrow 8 > 1 + 7$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 3 бетағйир гузошта мешавад.

3) $\text{Label}[5] > \text{Label}[1] + \text{Graf}[1][5] \Rightarrow 15 > 1 + 15$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 5 бетағйир гузошта мешавад.

Ба ҳамин монанд дар массиви Label ҳеч тағйирот ба вуҷуд намеояд, вале массиви Active шакли зеринро мегирад:

Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	0	0	1	1	1	0

Амали ҷустуҷӯро аз қуллаи аз назаргузарондашудаи дорои қимати хурдтарин оғоз мекунем, ин қуллаи 3-юм мебошад. Аз ин қуллаи ду теға (3,1) ва (3,4) мебарояд. Барои ин қуллаҳо низ шарти $\text{Label}[j] > \text{Label}[i] + \text{Graf}[i][j]$ -ро месанҷем.

1) $\text{Label}[1] > \text{Label}[3] + \text{Graf}[3][1] \Rightarrow 1 > 8 + 7$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 1 бетағйир гузошта мешавад.

2) $\text{Label}[4] > \text{Label}[3] + \text{Graf}[3][4] \Rightarrow 12 > 8 + 11$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 4 бетағйир гузошта мешавад.

Ба ҳамин монанд дар массиви Label ҳеч тағйирот ба вуҷуд намеояд, вале массиви Active шакли зеринро мегирад:

Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	0	0	0	1	1	0

Амали ҷустуҷӯро аз қуллаи аз назаргузарондашудаи дорои қимати хурдтарин оғоз мекунем, ин қуллаи 4-юм мебошад. Аз ин қуллаи се теға (4, 2), (4, 3), (4, 6) мебарояд. Барои ин қуллаҳо низ шарти $\text{Label}[j] > \text{Label}[i] + \text{Graf}[i][j]$ -ро месанҷем.

1) $\text{Label}[2] > \text{Label}[4] + \text{Graf}[4][2] \Rightarrow 4 > 11 + 8$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 2 бетағйир гузошта мешавад.

2) $\text{Label}[3] > \text{Label}[4] + \text{Graf}[4][3] \Rightarrow 8 > 11 + 11$. Аз сабаби оне, ки шарт иҷро намегардад, бинобар ин, қуллаи 3 бетағйир гузошта мешавад.

4) $\text{Label}[6] > \text{Label}[4] + \text{Graf}[4][6] \Rightarrow 32767 > 11 + 12$. Пас аз ин ҷо $\text{Label}[6] = \text{Label}[4] + \text{Graf}[4][6] = 11 + 12 = 23$.

Ба ҳамин монанд дар массиви Label ҳеч тағйирот ба вуҷуд намеояд, вале массиви Active шакли зеринро мегирад:

Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	0	0	0	0	1	1

Амали ҷустуҷӯро аз қуллаи аз назаргузарондашудаи дорои қимати хурдтарин оғоз мекунем, ин қуллаи 5-юм мебошад. Аз ин қуллаи се теға (5,1), (5,2), (5,6) мебарояд. Барои ин қуллаҳо низ шарти $\text{Label}[j] > \text{Label}[i] + \text{Graf}[i][j]$ -ро месанҷем.

1) $Label[1] > Label[5] + Graf[5][1] \Rightarrow 1 > 15 + 14$. Аз сабаби оне, ки шарт ичро намегардад, бинобар ин, қуллаи 1 бетағйир гузошта мешавад.

2) $Label[2] > Label[5] + Graf[5][2] \Rightarrow 4 > 15 + 13$. Аз сабаби оне, ки шарт ичро намегардад, бинобар ин, қуллаи 1 бетағйир гузошта мешавад.

5) $Label[6] > Label[5] + Graf[5][6] \Rightarrow 23 > 15 + 8$. Аз сабаби оне, ки шарт ичро намегардад, бинобар ин, қуллаи 6 бетағйир гузошта мешавад.

Пас аз ин амал массивҳои мо шакли зеринро мегиранд:

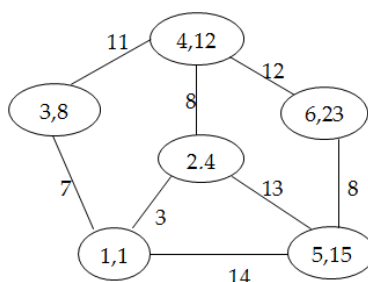
Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	0	0	0	0	0	1

Массиви Label

Рақами тартибӣ	1	2	3	4	5	6
Элемент	1	4	8	11	15	23

Пас аз ин амал графамон шакли зеринро мегирад.



Амали ҷустуҷӯро аз қуллаи аз назаргузарондашудаи дорои қимати хурдтарин оғоз мекунем, ин қуллаи 6-юм мебошад. Аз ин қуллаи ду тӯға (6,4) ва (6,5) мебарояд. Барои ин қуллаҳо низ шарти $Label[j] > Label[i] + Graf[i][j]$ -ро месанҷем.

3) $Label[4] > Label[6] + Graf[6][4] \Rightarrow 11 > 23 + 12$. Аз сабаби оне, ки шарт ичро намегардад, бинобар ин, қуллаи 4 бетағйир гузошта мешавад.

4) $Label[5] > Label[6] + Graf[6][5] \Rightarrow 15 > 23 + 8$. Аз сабаби оне, ки шарт ичро намегардад, бинобар ин, қуллаи 5 бетағйир гузошта мешавад.

Ба ҳамин монанд дар массиви Label ҳеҷ тағйирот ба вуҷуд намеояд, вале массиви Active шакли зеринро мегирад:

Массиви Active

Рақами тартибӣ	1	2	3	4	5	6
Элемент	0	0	0	0	0	0

Аз сабаби оне, ки дар массиви Active ягон элементи ба 1 баробар боқӣ намонд, бмноба ин, алгоритм дар ин ҷо қорашро ба анҷом мерасонад ва роҳи қутоҳтарин баробар аст ба $Label[4]=11$.

```
#include <iostream>
using namespace std;
int min(int x, int y){
    return ((x)<(y)?(x):(y));
}
int mass[100][100]; // массиви вазн
int i,j,N,M,W;
int a,b,c;
```

```

int start,end;
//массиве, ки дар он роҳи кутохтарин нигоҳ дошта мешавад (Label).
int D[100];
int used[100];
int short_way (int start, int end);
int main() {
    cin>>N>>M; // миқдори теға ва қуллаҳои граф
    //дар аввал матритса ро бо ададҳои калон пур мекунем
    for ( i = 1 ; i <=N ; i++)
        for ( j =1 ; j <= N ; j++) {
            mass[i][j]=32767;
            mass[j][i]=32767;
        }
    //дохил намудани рақами теғаҳо ва масофаи байни онҳо.
    for (i = 1; i <= M; i ++ ) {
        cin>>a>>b>>c;
        mass[a][b]=c;
        mass[b][a]=c;
    }
    //Хориҷ намудани матритсаи ҳамсоягӣ.
    cout<<" Matrix hamsoya:"<<endl;
    for ( i = 1 ; i <= N ; i++){
        for ( j = 1 ; j <= N ; j++)
            cout<<mass[i][j]<<" ";
        cout<<endl;
    }
    cout<<"Enter the first and the last point: ";
    cin>>start>>end;
    cout<<"\nThe shortest way is: \n"<<short_way(start,end) ;
    getch ();
    return 0;
}
//Зербарнома, барои ёфтани роҳи кутохтарин бо истифода аз
// алгоритми Дейскри
int short_way (int start, int end)
{
    /* Истифодаи алгоритми Дейскри */
    for ( i = 1 ; i <= N ; i++)
        used[i]=0
        for( i = 1 ; i <=N ; i++)
            D[i] = mass[start][i]; // массив барои нигоҳ доштани роҳи
            // кутохтарин аз қуллаи start то қуллаи i
            used[start] = 1; // ; // истодан дар аввали роҳ.
            for( i = 1 ; i <= N-2 ; i++)
            {
                int min_v = 1000000;
                for (j = 1; j <= N; j++ )
                    if (used[j]==0 && D[j]< min_v) /*агар ҳанӯз дар қуллаи
j ҳаракат накарда бошем ва қимати қуллаи j хурд аз қимати пешина
бошад, онгоҳ ба min_v қимати хурдтарин ва ба W рақами қуллаи дорои

```

```

қимати хурдтаринро мебахшем.*/
    {
        min_v = D[j];
        W = j;
    }
    used[W]=1;
    for( j = 1 ; j <= N ; j++)
        if (used[j]==0)
            // роҳи кутохтаринро интихоб мекунем.
            D[j]=min( D[j], D[W] + mass [W][j]);
    if (start == end)
        D[end]=0;
    return D[end];
}
return 0;
}

```

Намунаи ҳалли мисолҳо

1) Барномае нависед, ки пурра будани графро муайян кунад.

```

#include <iostream>
using namespace std;
int main(){
    int **graf;
    int n, k, i, j;
    cin>>n;
    graf=new int *[n];
    for( i=0; i<n; i++)
        graf[i]=new int[n];
    for( i=0; i<n; i++)
        for ( j=0; j<n; j++)
            graf[i][j]=0;
    for (k=0; k<n; k++)
    {
        cin>>i>>j;
        graf[i][j]=1;
    }
    k=0;
    for( i=0; i<n; i++)
        for ( j=0; j<n; j++)
            if (graf[i][j]!=0)
                k++;
    if (k==(n*(n-1)/2))
        cout<<"graf purra meboshad";
    else
        cout<<"graf nopurra meboshad";
    cout<<"graf eileri nameboshad";
    return 0;
}

```

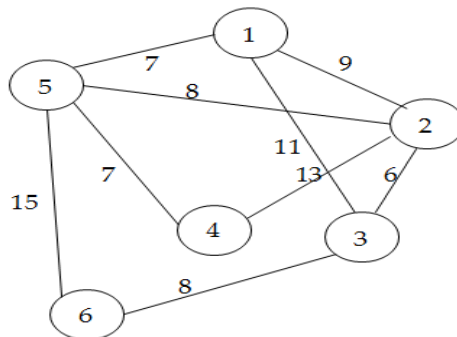
2) Барномае нависед, ки эйлери будани графро муайян кунад.

```

#include <iostream>
using namespace std;
int main(){
    int **graf;
    int n, k, i, j, one=0;
    cin>>n;
    graf=new int *[n];
    for( i=0; i<n; i++)
        graf[i]=new int[n];
    for( i=0; i<n; i++)
        for ( j=0; j<n; j++)
            graf[i][j]=0;
    for (k=0; k<n; k++)
    {
        cin>>i>>j;
        graf[i][j]=1;
    }
    k=0;
    for( i=0; i<n; i++)
        for ( j=0; j<n; j++)
            if (graf[i][j]==0)
                one++;
    if (one %2==0)
    {
        cout<<"graf eileri meboshad";
        return 0;
    }
    cout<<"graf eileri nameboshad";
    return 0;
}

```

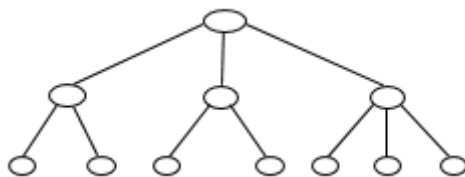
Графи зерин дода шудааст. Барномае нависед, ки дар он занҷири Эйлери муайян кунад.



Дарахтҳо

Дарахтҳо низ яке аз шаклҳои графӣ бе ҳалқа мебошанд, ки дар аксар ҷаҳонҳои компютерӣ қорбурди зиёд доранд. Бо суҳанҳои дигар графӣ алоқанок дарахт номида мешавад, агар барои он шарт $P-A=1$ иҷро градад. Дар ин ҷо P -миқдори қуллаҳо, A -миқдори теғҳо мебошад. Дилхоҳ графӣ дорои n

кулла, ҳатман $n-1$ -то тега дорад. Ҳар як гиреҳи дарахт метавонад дорои якчанд ишорагар бошад. Намунае аз дарахтҳо дар поён оварда шудааст.



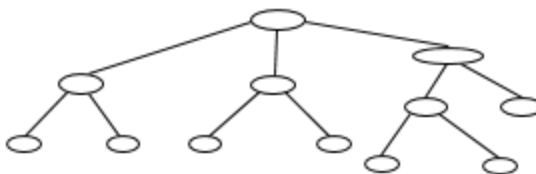
Дар ин ҷо ҳар як доира як гиреҳ ва ҳар як хат як пайванд (ишорагар) ном дорад. Дар ин дарахт баъзе гиреҳҳо дорои як пайванд ва баъзеашон дорои ду ва баъзеашон дорои се пайванд мебошанд.

Дарахтҳое, ки ҳар як гиреҳи онҳо дорои ду пайванд мебошад, дарахтҳои дӯй номида мешаванд.

Дар ин ҷо мо фақат дарахтҳои дӯиро мавриди баҳс қарор медиҳем.

Дарахтҳои дӯй

Дарахтҳои дӯй дарахтҳое ҳастанд, ки ҳар гиреҳи он ҳадди аксар дорои ду пайванд мебошад. Асоси бисёр бозиҳои компютерӣ дар асоси ин навъ дарахтҳо сохта мешаванд. Намунаи дарахти дӯй шакли зеринро дорад:

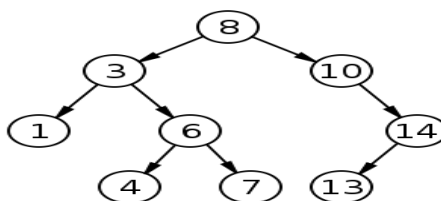


Дарахтҳои компютерӣ баракси дарахтҳои маъмулӣ мебошанд, бинобар ин реша дар боло нишон дода мешавад. Аз ин ҷост, ки аввалин гиреҳи дарахтро решаи дарахт меноманд. Яке аз намуди маъмули дарахтҳои дӯй ин дарахтҳои дӯии ҷустуҷӯӣ мебошанд. Дар дарахтҳои дӯй додаҳо аз рӯи қоидаи зерин нигоҳ дошта мешаванд:

- 1) Аввалин додони дохилкардашуда дар реша қарор мегирад;
- 2) Додаи дуюми бо реша муқоиса карда мешавад, чунончи агар аз реша калон бошад дар тарафи рости он, вагарна дар тарафи чапи он қарор мегирад.
- 3) Иттилооти баъдӣ монанди додаи дуввум бо реша муқоиса карда мешавад, чунончи агар аз он калон бошад, дар тарафи рости реша ва дар акси ҳол дар тарафи чап қарор мегирад.
- 4) Элементҳои баробар дар дарахти дӯй қарор намегиранд.

Ҳар гиреҳе, ки дорои ду шоха мебошад, ҳамчун реша фаҳмида мешавад. Барои ошноӣ бо ин шева, равиши дар дарахт нигоҳ доштани ададҳои зеринро дида мебароем.

8 10 3 6 14 13 1 4 7



Амали гузоштани ададҳо дар дарахти дӯй аз ин расм баръало маълум буда, ҳочат ба шарҳу тавзеҳ нест.

Сохтори гиреҳи дарахти дӯй

Ҳар гиреҳи дарахти дӯй монанди гиреҳи листи дупайвандӣ аз як баҳши иттилоот ва ду баҳши ишорагар ташкил шудааст, ки як пайванд ба зердарахти тарафи чап ва як пайванд ба зердарахти тарафи рост ишорага мекунад. Сохтори умумии як гиреҳи дарахти дӯй шакли зеринро дорад:

Дар ин ҷо `tree_name` –номи дарахт, `type`- дилхоҳ навъ ва `information_name` номи додаи вобаста ба `type` буда, `*left` ва `*right` ишорагари пайванди рост ва чап мебошанд.

```
struct tree_name {
    type information_name;
    tree_name *left;
    tree_name *right;
}
```

Мисол муарифии дарахтии дӯй барои ададҳои бутун:

```
struct tree {
    int data;
    tree *left;
    tree *right;
}
```

Пас аз муаррифии гиреҳи дарахти дӯй, зарурати барои он ҷудо намудани ҳофиза пеш меояд. Ин амал тавассути оператори `new` иҷро карда мешавад. Тавре ки қаблан зикр карда будем, аввалин гиреҳи дарахт реша махсуб шуда, гиреҳҳои баъди ба тарафи чап ва ё рост он қарор мегиранд. Бигзор `a` ва `b` ду ишорага аз навъи дарахти `tree` буда, `a` ба решаи дарахти ишора мекунад. Ин тағйирёбандаҳо ба сурати зерин ба ҳамдигар пайваст карда мешаванд.

```
a->right=NULL;
a->left=NULL;
if (b==NULL)
    b=a;
else if(a->data>b->data)
    b->right=a;
else
    b->left=a;
```

Мисол. Барномае нависед, ки амали (зербарномаи) гузоштан ва хориҷ намудани элементҳои дарахти дӯйро дора бошад.

```
#include <iostream>
#include<windows.h>
using namespace std;
struct TreeNode
{
    int Data; //қимати гиреҳ
    TreeNode *left; //ишорагар ба шохаҳои қаблӣ ва баъдӣ
    TreeNode *right;
};
```

```

//Иловакунии гиреҳ
void Add(int aData, TreeNode *&aTreeNode)
{
    //Агар шоха мавҷуд набошад, онро сохта ба он қимат мебахшем
    if (!aTreeNode)
    {
        aTreeNode = new TreeNode;
        aTreeNode->Data = aData;
        aTreeNode->left = 0;
        aTreeNode->right = 0;
        return;
    }
    else //дар ҳолати акс
        if (aTreeNode->Data>aData)
            { // Агар он қимати он аз қимати шохаи ҷори хурд бошад,
//дар тарафи чап навишта мешавад.
                Add(aData, aTreeNode->left);
            }
        else
            { //Дар ҳолати акс дар тарафи рост
                Add(aData, aTreeNode->right);
            };
};
}
void FreeTree(TreeNode *aTreeNode)
{
    if (!aTreeNode) return;
    FreeTree(aTreeNode->left);
    FreeTree(aTreeNode->right);
    delete aTreeNode;
    return;
}
//Барои мавқеи курсорро тағйир додан функсияи зеринро менависем
void gotoxy(short x, short y)
{
    COORD pos = {x,y};
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), pos);
}
//Хориҷкунии дарахт
void print_tree(TreeNode *idx, unsigned int pos, unsigned int pos_mod,
unsigned int y) {
    if(idx==NULL) return;
    gotoxy(pos, y);
    cout<<idx->Data<<endl;
    print_tree(idx->left, pos-pos_mod, pos_mod/2, y+1);
    print_tree(idx->right, pos+pos_mod, pos_mod/2, y+1);
}
//Функсияи асосӣ
int main()
{
    TreeNode *Root = 0;
}

```

```

int x[] = {8,10, 3, 6, 14, 13, 1, 4, 7};
for (int i = 0; i<9; i++)
{
    Add(x[i], Root);
}
print_tree(Root, 25, 10, 1);
FreeTree(Root);
return 0;
}

```

Натиҷаи барнома шакли зеринро дорад:



Дарахти ҷустуҷӯи дӯй

Барои ҷустуҷӯи гиреҳе дар дарахтҳои дӯй метавон аз ҳалқаҳои такрорӣ ё алгоритмҳои бозгашти истифода кард. Дар поён зарбарномаеро, ки дар он гиреҳи k-ум аз дарахти дӯй ҷустуҷӯ карда мешавад меорем.

```

.....
TreeNode *Treensearch(TreeNode *root, int k){
    TreeNode *curr = root;
    while (curr) {
        if (k == curr->Data)
            return curr;
        if (k < curr->Data)
            curr = curr->left;
        else
            curr = curr->right;
    }
    return 0;
}
//Функсияи асосӣ
int main()
{
    TreeNode *Root = 0;
    int k;
    cout<<"k="; cin>>k;
    int x[] = {8,10, 3, 6, 14, 13, 1, 4, 7};
    for (int i = 0; i<9; i++)
    {
        Add(x[i], Root);
    }
    TreeNode *h=Treensearch(Root, k);
    if (h!=NULL)
        cout<<h->Data;
    FreeTree(Root);
    return 0;
}

```


Чустучӯ (паёмиш)-и дарахти дӯй

Мақсади чустучӯи дарахт-ин азназаргузаронидани ҳамаи элементҳои дарахт мебошад. Роҳҳои гуногуни паёмиши дарахтҳои дӯй вучуд дорад, ки аз ҳама асоситаринашон дар поён оварда шудаанд:

- 1) Inorder;
- 2) Postorder;
- 3) Preorder.

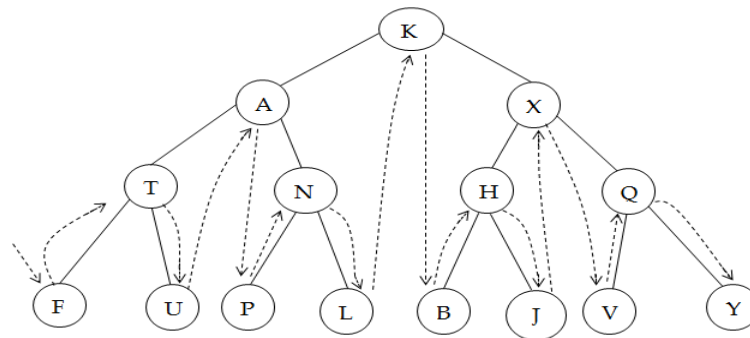
Ҳоло ҳар яке аз ин роҳҳо дар алоҳидагӣ дида мебароем.

Паёмиши inorder

Алгоритми ин паёмиш шакли зеринро дорад:

- 1) Азназар гузаронидани шохаҳои тарафи чап;
- 2) Аз назар гузаронидани реша;
- 3) Аз назар гузаронидани шохаҳои тарафи рост.

Бинобар ин амали ин паёмиш ба сурати зерин аст: аз реша оғоз карда, ба самти чап то расидан ба охири гирех ҳаракат карда мешавад. Пас аз расидан ба охири гирех муҳтавои он навишта шуда, сипас ба тарафи рост ҳаракат карда мешавад. Пас аз расидан ба қуллаи тарафи рост онро ҳамчун реша ба назар гирифта, ба самти чапи он ҳаракат идома дода мешавад. Пас аз расидан ба охири гирехи самти чап, муҳтавои он навишта шуда, боз ба самти рост ҳаракат карда мешавад. Ин раванд то мулоқот кардани ҳамаи гирехҳо идома дода мешавад. Барои моҳияти ин амалро пурра дарк кардан, дар поён ба намуди графикӣ оварда шудааст.



Натиҷаи амали паёмиши inorder дар ин дарахт чунин аст:
FTUARNLKBVHJXVQY.

Амали ин навъ паёмиш ба намуди бозгаштӣ анҷом дода мешавад. Аз ин рӯ барои иҷрои ин амал ба намуди бозгаштӣ функсияи ба шакли зерин навишта мешавад:

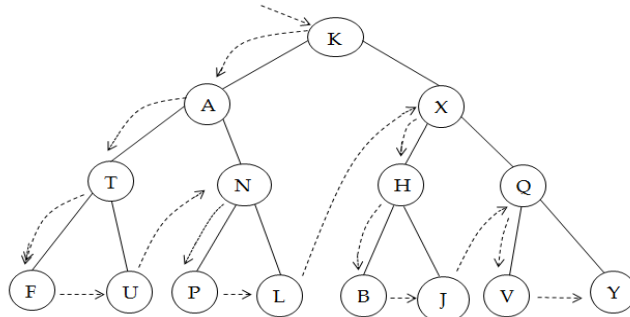
```
void inorder(TreeNode *root) {
    if (root == NULL) // Ҳолати базавӣ
    {
        return;
    }
    inorder(root->left);
    cout<<root->Data<<" ";
    inorder(root->right);
}
```

Паёмиши postorder

Алгоритми ин паёмиш шакли зеринро дорад:

- 1) Мулоқоти реша;
- 2) Мулоқоти шохаҳои тарафи чап;
- 3) Мулоқоти шохаҳои тарафи рост.

Аз расми зерин амали ин навъ паёмиш аён буда, ҳочат ба шарҳу тавзеҳ намебошад.



Натиҷаи амали паёмиши preorder дар ин дарахт чунин аст:

KATFUNPLXHNBQVY

Амали ин навъ паёмиш ба намуди бозгаштӣ анҷом дода мешавад. Аз инрӯ барои иҷрои ин амал ба намуди бозгаштӣ функсияи ба шакли зерин навишта мешавад:

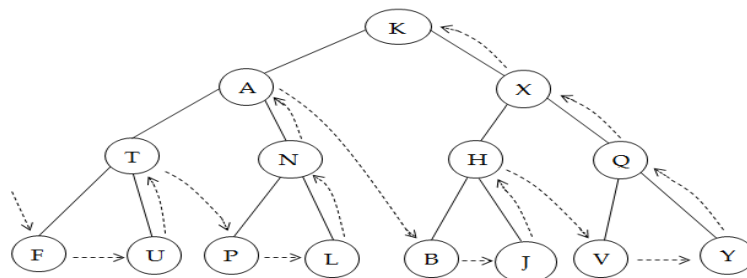
```
void preorder(TreeNode *root){
    if (root == NULL) // Ҳолати базавӣ
    {
        return;
    }
    cout<<root->Data<<" ";
    preorder(root->left);
    preorder(root->right);
}
```

Паёмиши postorder

Алгоритми ин паёмиш шакли зеринро дорад:

- 1) Мулоқоти зердарахтҳои тарафи чап;
- 2) Мулоқоти шохаҳои тарафи рост.
- 3) Мулоқоти реша;

Аз расми зерин амалкарди ин алгоритм аён буда, ҳочат ба шарҳу тавзеҳ нест.



Натиҷаи амали паёмиши postorder дар ин дарахт чунин аст:
FUTPLNABJHVVYQXK

Амали ин навъ паёмиш ба намуди бозгаштӣ анҷом дода мешавад. Аз инрӯ барои иҷрои ин амал ба намуди бозгаштӣ функсияи ба шакли зерин навишта мешавад:

```
void postorder(TreeNode *root) {
    if (root == NULL) // Ҳолати базавӣ
    {
        return;
    }
    postorder(root->left);
    postorder(root->right);
    cout<<root->Data<<" ";
}
```

Мисол. Барномае нависед, ки миқдори такроршавии ҳар як калимаи матни додашударо ҳисоб кунад.

Азбаски руйҳаи калимаҳо пешкаи маълум нест, бинобар ин, наметавон пешаки онҳоро мураттаб кард. Агар усули хаттии ҷустуҷӯиро истифода кунем, барнома хело суст кор мекунад.

Яке аз усулҳои сода- мураттаб нигоҳ доштани калимаҳои алақай ба даст омада мебошад. Барои ин, аз дарахтҳои дӯй истифода мекунем, яъне мисоли қаблан навиштаамонро мукамал мекунем. Ҳар як шоҳаи дарахт дорои элементҳои зерин мебошад:

- ишорагар ба матни калима;
- ҳисобкунаки миқдори такроршавии калима;
- ишорагар ба шоҳаи чап;
- ишорагар ба шоҳаи рост.

Мисол. Миқдори такроршавии ҳар як калимаи ду мисраъ шеъри зеринро муайян карда, натиҷаро ба намуди дарахти дӯй тасвир мекунем.

odami odam nabinad az kujo odam shavad,
sham` agar otash nabinad az kuzo ravshan shavad.

Матни барнома:

```
#include <iostream>
#include<windows.h>
using namespace std;
struct TreeNode{
    char *Data; //ишорагар ба сатр (калима)
    int count; //қимати гиреҳ
    TreeNode *left; //ишорагар ба шоҳаҳои қаблӣ ва баъд?
    TreeNode *right;
};
// Функсия барои иловакунии гиреҳ ба дарахт
TreeNode *addTree(TreeNode *p, char *w) {
    int cond;
    if(p == NULL) {
        p = new TreeNode[sizeof(TreeNode)];
        p->Data= strdup(w);
```

```

        p->count = 1;
        p->left = p->right = NULL;
    } else if((cond = strcmp(w, p->Data)) == 0)
        p->count++;
    else if(cond < 0)
        p->left = addTree(p->left, w);
    else
        p->right = addTree(p->right, w);
    return p;
}
//Озод кардани ҳофиза
void FreeTree(TreeNode *aTreeNode)
{
    if (!aTreeNode) return;
    FreeTree(aTreeNode->left);
    FreeTree(aTreeNode->right);
    delete aTreeNode;
    return;
}
//Барои мавқеи курсорро тағйир додан функсияи зеринро //менависем
void gotoxy(short x, short y)
{
    COORD pos = {x,y};
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), pos);
}
//Хориҷкунии дарахт
void print_tree(TreeNode *idx, unsigned int pos, unsigned int pos_mod,
unsigned int y) {
    if(idx==NULL) return;
    gotoxy(pos, y);
    cout<<idx->Data<<"="<<idx->count<<endl;
    print_tree(idx->left, pos-pos_mod, pos_mod/2, y+1);
    print_tree(idx->right, pos+pos_mod, pos_mod/2, y+1);
}
//Функсияи асосӣ
int main()
{
    TreeNode *root;
    char word[MAXWORD];
    root=NULL;
    do {
        cin>>word;
        if(isalpha(word[0]))
            root = addTree(root, word);
    }while(word[0]!='0'); // нишонаи ба охири раси 0 мебошад
    print_tree(root, 40, 20, 1);
    FreeTree(root);
    return 0;
}

```

Натиҷаи кори барнома:

```

                                odami=1
                                shavad,=1
                                shavad.=1
nabinad=2
                                sham'=1
                                otash=1
                                ravshan=1
az=2
agarkujo=1
kuzo=1

```

Мисоли 2. Барномае нависед, ки элементҳои дарахти дӯйро мураттаб намояд.

```

#include <iostream>
using namespace std;
#include <ctype.h>
struct TreeNode{
    int Data;
    TreeNode *left; //ишорагар ба шохаҳои қаблӣ ва баъдӣ
    TreeNode *right;
};
struct TreeNode * addTree(int x, TreeNode *tree) {
    if (tree == NULL) // Агар дарахт мавҷуд набошад, шохаҳояшро месозем
    {
        tree = new TreeNode; //ҳофиза барои гиреҳ
        tree->Data = x; //майдон барои қимат
        tree->left = NULL;
        tree->right = NULL; //ба шохаҳо аввал қимати холӣ мебахшем
    }
    else // дар ҳолати акс
        if (x < tree->Data) //агар элементҳои x аз элементҳои реша
//хурд бошад, ба тарафи чап меравем.
            tree->left = addTree(x, tree->left);
//Ба таври рекурсивӣ элемент илова мекунем.
            else //вагарна ба тарафи рост меравем
                tree->right = addTree(x, tree->right);
// Ба таври рекурсивӣ элемент илова мекунем.
        return tree;
    }
// хориҷқунии элементҳои дарахт
void treeprint(TreeNode *tree){
    if (tree != NULL) { //то расидан ба шохаи холӣ
        treeprint(tree->left);
//Ба таври рекурсивӣ хориҷ кардани зершохаи чап
        cout << tree->Data << " "; // Намоиш додани решаи зер шоха
        treeprint(tree->right);
// Ба таври рекурсивӣ хориҷ кардани зершохаи рост
    }
}
void FreeTree(TreeNode *aTreeNode){
    if (!aTreeNode) return;
    FreeTree(aTreeNode->left);
    FreeTree(aTreeNode->right);
    delete aTreeNode;
}

```

```

    return;
}
//Функсияи асосӣ
int main(){
    TreeNode *root=NULL;
    int n, t;
    cin>>n;
    for (int i=0; i<n; i++){
        t=rand()%25+1;
        root = addTree(t, root);
    }
    treeprint(root);
    FreeTree(root);
    return 0;
}

```

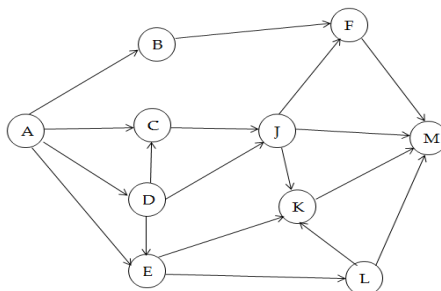
Саволҳо барои мустаҳкамкунӣ

- 1) Граф чист?
- 2) Графҳо дар кучо истифода бурда мешаванд?
- 3) Масъалаи назарияи графҳо кай пайдо шуд?
- 4) Граф чанд хел мешавад?
- 5) Графи Эйлери чи гуна граф аст?
- 6) Кадом намуди алгоритмҳои татқиқи графҳо медонед?
- 7) Алгоритми Дейскри чи гуна алгоритм аст?
- 8) Чустучӯ дар умқи аз чустучӯ дар васеъгӣ чи фарқият дорад?
- 9) Дараҷаи граф чист?
- 10) Занҷир гуфта чиро меноманд?
- 11) Дарахт гуфта чиро меноманд?
- 12) Дарахт аз граф чӣ фарқият дорад?
- 13) Оё метавон элементҳои додасударо дар дарахти дӯй чустучӯ кард?
- 14) Дарахтҳои дӯй чи гуна дарахтанд?
- 15) Паёмиш чист ва чанд навъи паёмиши дарахтҳои дӯй вучуд дорад?
- 16) Оё метавон, амали мураттабсозии додахоро тавасути дарахтҳои дӯй анҷом дод?

Мисолҳо барои кори мустақилона

- 1) Функсияе нависед, ки суммаи элементҳои дарахти дӯй ро ҳисоб кунад.
- 2) Барномае нависед, ки дар дохили элементҳои дарахти дӯй қимати додасударо чустучӯ кунад.
- 3) Барномае нависед, ки ду дарахти дӯйро муқоиса кунад.
- 4) Барномае, нависед, ки баландии максималии дарахтро муайян кунад.
- 5) Матне дода шудааст. Барномае нависед, ки миқдори такрор-шавии ҳар як калимаи онро муайян кунад.
- 6) Барномае нависед, ки аз дарахти дӯй ҳамаи элементҳои тоқро нест кунад.
- 7) Барномае нависед, ки миқдори элементҳои палиндроми дарахти дӯйро ҳисоб кунад.

8) Дар расми зерин системаи шахрҳои A, B, C, D, E, F, J, H, K, L, M дода шудааст. Муайян кунед, аз шахри A то шахри L чанд роҳ мавҷуд аст.



9) Дар байни маҳаллаҳои аҳолинишини A, B, C, D, E роҳ мавҷуд аст, ки дарозии роҳҳо дар чадвали зерин оварда шудаанд. Муайян кунед, ки кадом шахр ба шахри A наздиктар воқеъ гардидааст.

	A	B	C	D	E	F
A	0	2	4	0	0	0
B	2	0	1	0	7	0
C	4	1	0	3	4	0
D	0	0	3	0	3	0
E	0	7	4	3	0	2
F	0	0	0	0	2	0

10) Дар байни маҳаллаҳои аҳолинишини A, B, C, D, E роҳ мавҷуд чадвали зерин оварда дарозтарини аз шахри A то шахри E ро муайян кунед.

маҳаллаҳои аҳолинишини аст, ки дарозии роҳҳо дар шудаанд. Роҳи

	A	B	C	D	E
A	0	0	3	1	0
B	0	0	4	0	2
C	3	4	0	3	2
D	1	0	0	0	0
E	0	2	2	0	0

11) Дар олимпиада N иштирокчӣ вучуд дорад, ки якчандтои онҳо бо ҳам шинос мебошанд. Оё метавон онҳоро ғаримустақим (опосредованно) бо ҳамдигар шинос кард?

12) N ҳалқаи бо ҳамдигар пайваст дода шудааст. (матритсаи $A[n][n]$ дода шудааст, ки $A[i][j]=1$ аст агар ҳалқаҳои i ва j бо ҳамдигар пайваст бошад ва дар акси ҳол $A[i][j]=0$ мебошад). Миқдори минималии ҳалқаҳоро нест кунед, ки аз ҳалқаҳои боқимонда занҷир ҳосил шавад.

13) Гурӯҳ аз N студент иборат мебошад, ки ҳар яки онҳо дорои $N/2$ дӯст буда, на зиёда аз K душман дорад. Яке аз онҳо дорои китобе мебошад, ки ҳама хоҳиши онро хонда, сипас бо якчанд нафари боқимонда баҳс карданро дорад.

Барномае нависед, ки:

1) Усули тақсим кардани китобро барои ҳамаи донишҷӯён муайян кунед, ки ба дасти ҳар яки онҳо танҳо як бор расида, охир ба соҳибаш баргардад.

2) Донишҷӯёнро ба S гуруҳ тавре тақсим кунед, ки китобро бо ҳам баҳс кунанд ва дар байни онҳо на зиёда аз P душман вучуд дошта бошад.

Эзоҳ: фарз карда мешавад, ки $S \cdot P \geq K$ аст.

14) N одам ва чадвали квадратшакли $A[N][N]$ дода шудааст. Агар элементи $A[i][j]$ ба 1 баробар бошад, маънои онро доад, ки одами i бо одами j шинос аст, $A[i][j] = A[j][i]$. Оё метавон одамонро бо ду гуруҳ тавре тақсим кард, ки дар ҳар ду гуруҳ танҳо одамони ношинос вучуд дошта бошанд.

15) 20 донишҷӯйро ба ду гуруҳ тавре тақсим кунед, ки шумораи онҳо аз ҳамдигар на зиёда аз 2 маротиба фарқ кунад, агар маълум бошад, ки дар дилхоҳ гуруҳ танҳо донишҷӯёни бо ҳамдигар шинос вучуд дорад. Давраи шиносӣ дар намуди матритсаи $x[20][20]$ дода мешавад, ки элементҳои шакли зеринро дорад.

$$x_{ij} = \begin{cases} 1, & \text{агар донишҷӯи } i \text{ бо донишҷӯи } j \text{ шинос бошад,} \\ 0, & \text{дар дигар ҳолатҳо} \end{cases}$$

16) N лифофа (конверти) шакли росткунҷашакл ва N руқъаи (откреткаи) росткунҷаи андозаи гуногун вучуд дорад. Оё метавон ҳамаи руқъахоро дар конвертҳо ҷойгир кард, ки дар ҳар як лифофа танҳо як руқъа ҷойгир шавад.

Эзоҳ. Руқъаро наметавон қат кард ва ё бо таври кунҷи дар лифофа ниҳод. Масалан руқъаи андозаи тарафҳояш ба 5:1 баробар метавонад дар лифофаҳои андозаашон ба 5:1, 6:3, 4.3:4.3 баробар ҷойгир карда шавад, аммо наметавонад дар лифофаҳои андозаашон ба 4:1, 10:0.5, 4.2:4.2 баробар ҷойгир гардад.

17) N дастгоҳ гуногуни бо ҳамдигар пайвасте, ки дорои N коргар мебошанд дода шудааст. Матритсаи $C[N][N]$ дода шудааст, ки $C[i,j]$ маҳсули коргари i -юмро дар дастгоҳи j -юм ифода мекунад. Муайян кунед:

а) Дар кадом дастгоҳ коргар бояд кор кунад, ки маҳсулнокиаш бисёртар бошад.

б) ба а) монанд, аммо дастгоҳҳо параллел ҷойгир буда, амали якхела иҷро карда шавад.

18) N – то хона ва K – то роҳ дода шудааст. Хонаҳо 1 то N рақамгузорӣ карда шудаанд. Ҳар як роҳро се адад, ки дутоаш рақами хона ва яктоаш дарозии роҳро муайян мекунад. Дар ҳар хона як одам зиндагӣ мекунад. Ҷойи вохурии ҳамаи одамҳоро ёбед, ки масофа аз онҳо то ҳамаи хонаҳо кӯтоҳтарин бошад.

Агар нуқта дар роҳ хобад, онгоҳ рақами хона, охири роҳ ва масофаи хонаи якумро ифода мекунад. Агар нуқта ба хона мувофиқ ояд, онгоҳ рақами ин хонаро ифода мекунад.

Қайд: дарозии роҳ – адади бутуни мусбат мебошад.

Манбаъҳои истифодашуда

- 1) А. Осимӣ. Луғати русӣ-тоҷикӣ. Москва-1985 г, 1279 с.
- 2) Айвор Хортон. Visual C++ 2010: полный курс = Ivor Horton's Beginning Visual C++ 2010. — М.: Диалектика, 2010. — 1216 с.
- 3) Акулич И.Л. Глава 4. Задачи динамического программирования // Математическое программирование в примерах и задачах. — М.: Высшая школа, 1986. — 319 с.
- 4) Арабов М.К., Нуров И.Д. Основы программирования на языке PascalABC.NET. Учебное-методическое пособие. — Душанбе: РТСУ, 2018. - 218 с.
- 5) Архангельский А.Я. Программирование в C++ Builder 4. - М.: БИНОМ, 1999. - 928с.
- 6) Ахо А. В., Хопкрофт Дж., Ульман Дж. Д. Структуры данных и алгоритмы.: Пер. с англ.: - М.: Издательский дом "Вильямс", 2003. - 384 с.
- 7) Ашӯров Х. Комилов О. Маҷмӯи масъалаҳо аз программасозӣ. — Душанбе, 2013. — 68 с.
- 8) Баженова И.Ю. Visual C++5.0.-М.: Диалог - МИФИ, 1998.-272с..
- 9) Березин Б.И., Березин С.Б. Начальный курс С и С++. - М.: Диалог_мифи, 1996. - 288 с.
- 10) Бруно Бабэ. Просто и ясно о Borland C++: Пер. с англ. - Москва: БИНОМ, 1994. 400с.
- 11) Бьярн Страуструп. Язык программирования. С++. Ч.2. Пер.с англ. - Киев, ДиаСофт, 1993. - 296с.
- 12) Бьярне Страуструп "Программирование принципы и практика использования С++." "И.Д. Вильямс", 2011. - 1223 с.
- 13) В.А. Складов. Язык С++ и объектно-ориентированное программирование: Справочное издание. - Минск: Вышэйшая школа, 1997. 480с.
- 14) В.В. Подбельский. Язык С++: Учебное пособие. - Москва: Финансы и статистика, 1995. 560с.
- 15) В.Н. Пильщиков. Сборник упражнений по языку Паскаль. М., «Наука», 1989
- 16) Вирт Н. Алгоритмы и структуры данных. — Спб: Невский диалект, 2001 г. — 352 с.
- 17) Г. Шилдт. Самоучитель С++: Пер. с англ. - Санкт-Петербург: ВHV-Санкт-Петербург, 1998. - 620с.
- 18) Габасов Р., Кириллова Ф. М. Основы динамического программирования. — Мн.: Изд-во БГУ, 1975. — 262 с.
- 19) Герберт Шилдт. С: полное руководство, классическое издание = С: The Complete Reference, 4th Edition. — М.: Вильямс, 2010. — 704 с.
- 20) Герберт Шилдт. Полный справочник по С++ = С++: The Complete Reference. — 4-е изд. — М.: Вильямс, 2011. — 800 с.
- 21) Голуб А. И. С и С++. Правила программирования. — М: БИНОМ, 1996 г. — 272 с.
- 22) Гукин Д. Язык программирования Си для «чайников» = C For Dummies. — М.: Диалектика, 2006. — 352 с.
- 23) Гуннерсон Э. Введение в С#. Библиотека программиста. — СПб.: Питер, 2001. — 304 с.
- 24) Дейтел Х., Дейтел П. Как программировать на С++: Пер. с англ. - М.: Издательство БИНОМ, 1998 - 1024 с.
- 25) Джонс Р., Стюарт Я. Програмуем на Си/Пер. с англ. и предисл. М.Л. Сальникова, Ю.В. Сальниковой. - М.: Компьютер, ЮНИТИ, 1994. - 236 с.

- 26) Джосьютис Н. С++ Стандартная библиотека. Для профессионалов. СПб.: Питер, 2004. - 730 с.
- 27) Дональд Кнут Искусство программирования, том 1. Основные алгоритмы = The Art of Computer Programming, vol.1. Fundamental Algorithms. — 3-е изд. — М.: «Вильямс», 2006. — 720 с.
- 28) Дэвис Стефан Р. С++ для "чайников", 4-е издание.: Перев. с англ.: Уч. пос. - М.: Издательский дом "Вильямс", 2001. - 336 с.
- 29) Елманова Н.З. Комель С.П. Введение в Borland С++ Builder.-М.:Диалог - МИФИ, 2000.- 344с.
- 30) Емеличев В. А., Мельников О. И., Сарванов В. И., Тышкевич Р. И. Лекции по теории графов. М.: Наука, 1990. 384с. (Изд.2, испр. М.: УРСС, 2009. 392 с.).
- 31) Жилмер Дж. Дердж, Дэвид Р. Мюссер, Атул Сейни. С++ и STL: справочное руководство =STL Tutorial and Reference Guide: С++ Programming with the Standard Template Library. — 2-е издание. — М.: Вильямс, 2010. — 432 с.
- 32) Игошин В. И. Математическая логика и теория алгоритмов. — 2-е изд., стер. — М.: ИЦ «Академия», 2008. — 448 с.
- 33) Ирэ Пол. Объектно-ориентированное программирование с использованием С++: Пер. с англ. - Киев: НИИПФ ДиаСофт Лтд, 1995. 480с.
- 34) Йенсен К., Вирт Н. Паскаль: руководство для пользователя/пер. с англ. и предисл. Д. Б. Подшивалова. – М.: Финансы и статистика, 1989. – 255 с.
- 35) К. Джамса. Учимся программировать на языке С++: Пер. с англ. - Москва: Мир, 1997. 320с.
- 36) К.Рейсдорф, К.Хендерсон. Освой самостоятельно Borland С++ Builder / Пер.с англ. -М.: "Издательство БИНОМ", 1998. -704 с.
- 37) Касьянов В. Н., Евстигнеев В.А. Графы в программировании: обработка, визуализация и применение. — СПб.: БХВ-Петербург, 2003. — 1104 с.
- 38) Керниган Б., Пайк Р. Практика программирования.: Пер. с англ. - М.: Издательский дом "Вильямс", 2004. - 288 с.
- 39) Керниган Б., Ритчи Д. Язык программирования Си. — 2-е изд. — М.: Вильямс, 2007. — 304 с.
- 40) Керниган Б., Ритчи Д. Язык программирования Си. – Санкт-Петербург: Невский диалект, 2001. – 352 с.
- 41) Кирсанов М. Н. Графы в Maple. М.: Физматлит, 2007. — 168 с.
- 42) Кнут, Дональд Эрвин. Искусство программирования. Пер. с англ.: - М.: "Вильямс", 2007. Т. 1: Основные алгоритмы. - 3-е изд. - 2007. - 712 с.
- 43) Кнут, Дональд Эрвин. Искусство программирования. Пер. с англ.: - М.: Издательский дом "Вильямс", 2007. Т. 2: Получисленные алгоритмы. - 3-е изд. 2007. -828 с.
- 44) Кнут, Дональд Эрвин. Искусство программирования. Пер. с англ.: - М.: Издательский дом "Вильямс", 2007. Т. 3: Сортировка и поиск. 3-е изд. -2007. - 822 с.
- 45) Кораблев В. Самоучитель Visual С++ .NET. — СПб.: Питер; Киев: Издательская группа ВHV, 2004. — 528 с.
- 46) Кормен Т. М. и др. Часть VI. Алгоритмы для работы с графами // Алгоритмы: построение и анализ. -2-е изд. — М.: Издательский дом «Вильямс», 2006. — 1296 с.
- 47) Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ, 2-е издание.: Пер. с англ. - М.: Издательский дом "Вильямс", 2007. - 1290 с.

- 48) Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К. Глава 15. Динамическое программирование // Алгоритмы: построение и анализ = Introduction to Algorithms / Под ред. И. В. Красикова. — 2-е изд. — М.: Вильямс, 2005. — 1296 с.
- 49) Кочан С. Программирование на языке Си = Programming in C. — 3-е изд. — М.: Вильямс, 2006. — 496 с.
- 50) Культин Н.Б. С/C++ в задачах и примерах. - СПб.:БХВ-Петербург,2001.- 288 с.
- 51) Леен Аммерааль. STL для программистов на C++, пер. с англ. Леен Аммерааль – М.; ДМК Пресс, 1999. — 240 с.
- 52) М. Эллис, Б. Строуструп. Справочное руководство по языку C++ с комментариями. - Москва: Мир, 1992. 445 с.
- 53) М.Теллес. Borland C++ Builder: библиотека программиста. - СПб: Питер Ком, 1998. -512 с.
- 54) Майо Дж. C# Builder. Быстрый старт. — М.: Бином, 2005. — 384 с.
- 55) Муллошараф Арабов. Асосҳои барноманависӣ дар забони PascalABC.NET. Китоби дарсӣ. Душанбе. РТСУ - 2018. -302с.
- 56) Натан Гуревич, Ори Гуревич. Visual C++: Освой самостоятельно/ Пер. с англ.- М.: ЗАО "Издательство БИНОМ", 1998г.-624 с.
- 57) Оре О. Теория графов. М.: Наука, 1968. 336с.
- 58) Павловская Т.А. С/C++. Программирование на языке высокого уровня. / Т.А. Павловская. - СПб.: Питер, 2002. - 464 с.
- 59) Павловская Т. А. Паскаль. Программирование на языке высокого уровня. Учебник для вузов. — СПб.: Питер, 2003. — 393 с.
- 60) Паппас К., Мюррей У. Эффективная работа: Visual C++ .NET. — СПб.: Питер, 2002. — 816 с.
- 61) Пашенков В.В. Язык программирования Си. - М.: Центр НТТМ "Алгоритм", 1990. - 76 с.
- 62) Першиков В.И., Савинков В.М. Толковый словарь по информатике.-М.: Финансы и статистика, 1991. - 543 с.
- 63) Подбельский В. В. 6.9. Шаблоны функций //Глава 6. Функции, указатели, ссылки // Язык Си++ / рец. Дадаев Ю. Г.. — 4. — М.: Финансы и статистика, 2003. — С. 230-236. — 560 с.
- 64) Подбельский В.В., Фомин С.С. Программирование на языке Си: Учеб. пособие. - М.: Финансы и статистика, 1998. - 600 с.
- 65) Понамарев В. А. Программирование на C++/C# в Visual Studio .NET 2003. Серия «Мастер программ». — СПб.: БХВ-Петербург, 2004. —352 с.
- 66) Порублев Илья Николаевич, Ставровский Андрей Борисович. Алгоритмы и программы. Решение олимпиадных задач. — М.: «Вильямс», 2007. — 480 с.
- 67) Прата С. Язык программирования C: Лекции и упражнения = C Primer Plus. — М.: Вильямс, 2006. — 960 с.
- 68) Седжвик Р. Фундаментальные алгоритмы на C++. Ч. 5: Алгоритмы на графах. - СПб.: "ДиаСофтЮП", 2002. - 496 с.
- 69) Седжвик Р. Фундаментальные алгоритмы на C++. Части 1-4: Анализ/Структуры данных/Сортировка/Поиск: Пер. с англ. - К.: Издательство "ДиаСофт", 2001. - 688 с.
- 70) Скляр В.А. Программирование на языках Си и Си++: Практ. пособие. - М.: Высш. шк., 1996. -240 с.
- 71) Стандарт C++ ISO / IEC 14882 : 1998(E).

- 72) Стенли Б. Липпман. С++ для начинающих: Пер. с англ. 2тт. - Москва: Унитех; Рязань: Гэлион, 1992, 304-345с.
- 73) Стефенс Д. Р. С++. Сборник рецептов. — КУДИЦ-ПРЕСС, 2007. — 624 с.
- 74) Стивен Прата Язык программирования С++ (С++11). Лекции и упражнения, 6-е издание— М.: «Вильямс», 2012. — 1248 с.
- 75) Страуструп Б. Программирование: принципы и практика использования С++, исправленное издание. — М.: Вильямс, 2011. — 1248 с.
- 76) Страуструп Б. Язык программирования С++. Специальное издание = The С++ programming language. Special edition. — М.: Бином-Пресс, 2007. — 1104 с.
- 77) Т. Сван. Освоение Borland С++ 4.5: Пер. с англ. - Киев: Диалектика, 1996. 544 с.
- 78) Т. Фейсон. ООП на Borland С++ 4.5: Пер. с англ. - Киев: Диалектика, 1996. 544с.
- 79) У. Сэвитч. С++ в примерах: Пер. с англ. - Москва: ЭКОМ, 1997. 736с.
- 80) Уилсон Р. Введение в теорию графов. Пер с англ. М.: Мир, 1977. 208с.
- 81) Уинер Р. Язык Турбо Си: Пер. с англ. -М.: Мир, 1991. - 384 с.
- 82) Франка П. С++: учебный курс. - СПб.: Питер, 2001. - 528 с.
- 83) Х. Дейтел, П. Дейтел. Как программировать на С++: Пер. с англ. - Москва: ЗАО "Издательство БИНОМ", 1998. 1024 с.
- 84) Холзнер С. Visual С++ 6: учебный курс - СПб: Питер,2001. - 576 с.
- 85) Холзнер С. Microsoft Visual С++5 с самого начала: 14 уроков для начинающих. - СПб.: Питер,1998.-480с.
- 86) Чумаев Э.Х. Асосхои информатика ва программасозй дар забони Паскал. Васоити таълими. Душанбе, 2002. - 75 с.
- 87) Шамис В.А. Borland С++ Builder: Программирование на С++ без проблем. - М.: Нолидж, 1997. – 266 с.
- 88) Шилдт, Герберт. Полный справочник по С, 4-е издание. : Пер. с англ. - М.: Издательский дом "Вильямс", 2002. - 704 с.
- 89) Щербина О. А. Методологические аспекты динамического программирования // Динамические системы, 2007, вып. 22. — с.21-36.
- 90) Юникод – Википедия [Электронный ресурс]. – Режим доступа: <http://ru.wikipedia.org/wiki/Unicode>
- 91) American National Standard for Information Systems - Programming Language C, X3.159-1989
- 92) Axel-Tobias Schreiner. OOP with ANSI-C. -Hanser, 2011. — 223 p.
- 93) Bertele U., Brioshi F. Nonserial dynamic programming. — N.Y.: Academic Press, 1972. — 235 pp.
- 94) Bjarne Stroustrup and Jonathan Shopiro: A Set of C classes for Co-Routine Style Programming. Proc. USENIX C++ conference, Santa Fe. November 1987. pp.417-439.
- 95) <http://progopedia.ru/>
- 96) <https://github.com/>
- 97) <https://habrahabr.ru>
- 98) <http://cppstudio.com/>
- 99) OEM 866 [Электронный ресурс]. – Режим доступа: <http://msdn.microsoft.com/ru-ru/goglobal/cc305166>.
- 100) The Unicode Consortium [Электронный ресурс]. – Режим доступа: <http://www.unicode.org>

Ба чопхона 20.02.2018 . супорида шуд. Ба чопаш 22.02.2018 .

имзо шуд. Андозаи 60x84 1/16. Ҷузъи чопӣ 26.

Адади нашр 300. Китоб дар чопхонаи

«ДСРТ» нашр шудааст. 734025,

ш. Душанбе