

Российско-Таджикский (славянский) университет

Практикум по программированию

**учебное пособие для учебное пособие для студентов – бакалавров и
магистрантов, изучающих информационные
технологии и программирование.**

Душанбе-2019

УДК 004..42 (046)

A79

ББК 32.973+74.202.4+92Я2

A-73

Арабов М.К., Замонов Б.М.

Практикум по программированию: учебное пособие для студентов – бакалавров и магистрантов, изучающих информационные технологии и программирование /М.К. Арабов, Б.М. Замонов Душанбе: РТСУ, 2019. 370 с.

В учебном пособии рассмотрены основные способы программирования на самых востребованных и современных языках программирования, таких как: PascalABC.NET, C++, Python, C#, Java, Go, Kotlin и Visual Basic.NET. Данное учебное пособие можно использовать для приобретения практических навыков алгоритмизации задач и программирование на современных языках программирования. В нем приводятся основные приемы и методы программирования, большое число типовых алгоритмов для решения задач различных классов, а также примеры их использования. Ей можно пользоваться при проведении практических занятий, дисциплин, связанных с программированием: «Практикум по программированию», «Основы алгоритмизации и языки программирования», «Теории алгоритмов», «Методы программирования», «Основы информатики и программирования», «Программная инженерия» и т.п.

Ответственный редактор:

к.ф.-м.н, Замонов М.З., доцент кафедры
информатики и ИС, РТСУ,

Рецензенты

д.ф.-м.н., Нуров И.Д., профессор кафедры
кафедры ИКТ, ТНУ

к.ф.-м.н, Кабилов М.М., заведующий кафедры
информатики и ИС, РТСУ, доцент

Рекомендовано к печати УМС РТСУ
Рекомендовано к печати РИС РТСУ

© Арабов М.К., Замонов Б.М. – 2019.

Содержание

Предисловие	6
1. Ввод и вывод данных, оператор присваивания: группа Begin	8
Задачи для самостоятельной работы	17
2. Целые числа: группа Integer	20
Задачи для самостоятельной работы	26
3. Логические выражения: группа Boolean	29
Задачи для самостоятельной работы	37
4. Условный оператор: группа If	41
Задачи для самостоятельной работы	55
5. Оператор выбора: группа Case	58
Задачи для самостоятельной работы	73
6. Цикл с параметром: группа For	74
Вложенные циклы	85
Задачи для самостоятельной работы	88
7. Цикл с условием: группа While	93
Задачи для самостоятельной работы	107
8. Последовательности: группа Series	110
Вложенные циклы	115
Задачи для самостоятельной работы	119
9. Процедуры и функции: группа Proc	121
1. Процедуры с числовыми параметрами	121
2. Функции с числовыми параметрами	126
3. Дополнительные задания на процедуры и функции	129
Задачи для самостоятельной работы	133
10. Минимумы и максимумы: группа Minmax	139
Задачи для самостоятельной работы	146
11. Одномерные массивы: группа Array	148
1. Формирование массива и вывод его элементов	148
2. Анализ элементов массива	154
3. Работа с несколькими массивами	160
4. Преобразование массива	161
а) Изменение элементов массива	161

b) Удаление и вставка элементов.....	163
c) Сортировка массива	166
5. Серии целых чисел	167
6. Множества точек на плоскости	171
Задачи для самостоятельной работы.....	174
12. Двумерные массивы (матрицы): группа Matrix	182
1. Формирование матрицы и вывод ее элементов.....	182
2. Анализ элементов матрицы	188
3. Преобразование матрицы.....	193
4. Диагонали квадратной матрицы.....	200
Задачи для самостоятельной работы.....	205
13. Символы и строки: группа String.....	211
1. Символы и их коды. Формирование строк	211
2. Посимвольный анализ и преобразование строк. Строки и числа.....	215
3. Обработка строк с помощью стандартных функций. Поиск и замена...	219
4. Анализ и преобразование слов в строке.....	222
5. Дополнительные задания на обработку строк.....	228
Задачи для самостоятельной работы.....	233
14. Двоичные (типизированные) файлы: группа File	237
1. Основные операции с двоичными файлами.....	237
a) Создание файла, ввод и вывод его элементов.....	237
b) Преобразование файла.....	248
c) Работа с нетипизированными двоичными файлами.....	251
2. Обработка нескольких числовых файлов. Файлы-архивы.....	253
3. Символьные и строковые файлы.....	255
4. Использование файлов для работы с матрицами	259
Задачи для самостоятельной работы.....	262
15. Текстовые файлы: группа Text	268
1. Основные операции с текстовыми файлами.....	268
2. Анализ и форматирование текста	275
3. Текстовые файлы с числовой информацией.....	280
4. Дополнительные задания на обработку текстовых файлов	284
Задачи для самостоятельной работы.....	286

16. Составные типы данных в процедурах и функциях: группа Param.....	290
1. Одномерные и двумерные массивы	290
2. Строки	296
3. Файлы	299
4. Записи.....	302
Задачи для самостоятельной работы.....	305
17. Рекурсия: группа Recur.....	313
1. Простейшие рекурсивные алгоритмы	313
2. Разбор выражений.....	321
3. Перебор с возвратом.....	326
Задачи для самостоятельной работы.....	328
18. Указатели и динамические структуры данных: группа Pointer.....	331
1. Стеки	331
2. Очереди.....	338
3. Двусвязные списки	344
Задачи для самостоятельной работы.....	359
Литература	368

Предисловие

*Я всегда брался за новый язык, чтобы
овладеть новым рабочим орудием.
Мирче Элиаде¹*

Учебное пособие является частью комплекса, предназначенного для обучения основам программирования на его самых востребованных языках. Пособие включает дополнительный задачник, а также набор примеров решения, обеспечивающих полноценный и эффективный процесс изучения основ программирования.

На протяжении многих лет в различных вузах для подготовки будущих it-специалистов используются языки C++, C#. Они вполне логичны, понятны и гибки, также способствуют развитию хорошего стиля программирования. Кроме того, в средней школе основы кодирования преподают именно на базе C++ и Python. Однако в последнее время другие языки программирования, такие как Java, VB.NET, Python, Go и Kotlin, в свою очередь, также становятся популярными. Во многих вузах их стали внедрять в образовательный процесс. Большинство из вышеперечисленных языков имеют простые и мощные интегрированные среды разработки, поддерживающие технологию IntelliSense.

В настоящее время существует множество подходов к изучению программирования. Однако нельзя приступать к изучению на каком-либо языке, не освоив методы разработки алгоритмов. Авторы предприняли попытку создать прикладное учебное пособие по алгоритмизации и программированию. Насколько им это удалось, судить читателю.

Сотни задач, представленные в пособии, с подробным решением имеют важное практическое значение. Эти задачи можно также использовать для подготовки студентов и школьников к участию в олимпиадах.

Отметим, что при работе над этим учебным пособием была использована следующая литература [1-26], а именно [1-6] для авторов являются основными источниками задач.

В процессе изучения данного пособия предполагается, что читатель имеет первоначальные навыки работы на персональном компьютере на ОС Windows или Linux и знаком со школьным курсом математики.

¹ Мирча Элиаде (рум. *Mircea Eliade*; 13 марта 1907, Бухарест, Королевство Румыния — 22 апреля 1986, Чикаго, США) — румынский, французский и американский философ, философ культуры, религиовед, историк религий, этнограф и писатель.

Авторы надеются, что их труд поможет читателям не только понять фундаментальные аспекты программирования, но и оценить красоту и изящество его идей и результатов.

Авторы будут благодарны и признательны всем тем, кто пришлёт замечания и пожелания по адресу: 734025, Республики Таджикистан, г. Душанбе, ул. Мирзо Турсун-Заде, 30. Email: cool.araby@mail.ru.

1. Ввод и вывод данных, оператор присваивания: группа Begin

Все входные и выходные данные в заданиях этой группы являются вещественными числами.

1) Дана сторона квадрата a . Найти его периметр $P = 4 \cdot a$.

PascalABC.NET	C++
<pre> Var a, P: Real; begin ReadLn(a); P := 4 * a; WriteLn('P = ', P); end. </pre>	<pre> #include <iostream> using namespace std; int main() { double a,P; cin >> a; P=4*a; cout << "P -> " << P << endl; return 0; } </pre>
Python	C#
<pre> a=float(input("a->")) P=4*a print(P) </pre>	<pre> static void Main(string[] args) { int a, p; a = Convert.ToInt32(Console.ReadLine()); p = 4 * a; Console.Write(p); Console.ReadKey(); } </pre>
Java	GO
<pre> public static void main(String[] args) { int a,P; Scanner pa=new Scanner(System.in); a=pa.nextInt(); P=4*a; System.out.println(P); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("a=?") var a int fmt.Scanf("%d", &a) var p=4*a fmt.Println("P=", p) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextFloat() val p = 4*a println(p) } </pre>	
VB.NET	
<pre> Sub Main() Dim a, p As Single a = Convert.ToSingle(Console.ReadLine()) p = 4 * a Console.WriteLine(p) End Sub </pre>	

- 2) Дан диаметр окружности d . Найти ее длину $L = \pi \cdot d$. В качестве значения π использовать 3.14.

PascalABC.NET	C++
<pre> Var d, L: Real; begin ReadLn(d); L:=d*pi; WriteLn('L = ',L); ReadLn; end. </pre>	<pre> #include <iostream> using namespace std; int main(){ double d,L; cin >> d; L= d * 3.14 cout<<"L="<<L<< endl; return 0; } </pre>
Python	C#
<pre> d= float(input("d:")) L=3.14*d print(L) </pre>	<pre> static void Main(string[] args){ Double d, L, pi; d = Convert.ToDouble(Console.ReadLine()); pi = Math.PI; L = d * pi; Console.Write(L); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { double L,d; Scanner ld=new Scanner(System.in); d=ld.nextDouble(); L=Math.PI*d; System.out.println(L); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("d=?") var d float64 fmt.Scanf("%v", &d) var L=d*3.14 fmt.Println("L=", L) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var d=reader.nextFloat() val L=Math.PI*d println(L) } </pre>	

- 3) Даны два ненулевых числа. Найти сумму, разность, произведение и частное их модулей.

PascalABC.NET	C++
<pre> Var a, b,s,m,z,t: Real; begin ReadLn(a, b); s:= abs(a) + abs(b) ; m:= abs(a) - abs(b) ; z:= abs(a) * abs(b) ; t:= abs(a) / abs(b) ; </pre>	<pre> #include <iostream> using namespace std; int main() { double a, b,s,m,z,t; cin >> a >> b; s= abs(a) + abs(b) ; m= abs(a) - abs(b) ; z= abs(a) * abs(b) ; </pre>

<pre>WriteLn(s, ' ', m, ' ', z, ' ', t); end.</pre>	<pre>t= abs(a) / abs(b) ; cout << "S= " <<s<< endl; cout << "M=" <<m << endl; cout << "Z=" << z<< endl; cout << "T=" << t << endl; return 0; }</pre>
Python	C#
<pre>a=float(input("a:")) b=float(input("b:")) p=abs(a)+abs(b) s=abs(a)-abs(b) t=abs(a)*abs(b) v=abs(a)/abs(b) print(p, "", s, "", v, "", t)</pre>	<pre>public static void Main(string[] args) { int a, b, s, r, p, t; a = Convert.ToInt(Console.ReadLine()); b = Convert.ToInt(Console.ReadLine()); s = Math.Abs(a) + Math.Abs(b); r = Math.Abs(a) - Math.Abs(b); p = Math.Abs(a) * Math.Abs(b); t = Math.Abs(a) / Math.Abs(b); Console.WriteLine("S={0}, R={1}, P={2}, T={2}", s, r, p, t); Console.ReadKey(true); }</pre>
Java	
<pre>public static void main(String[] args) { int A, B,s,r,d,u; Scanner rd=new Scanner (System.in); a=rd.nextInt(); b=rd.nextInt(); s=Math.abs(a)+Math.abs(b); r=Math.abs(a)-Math.abs(b); d=Math.abs(a)/Math.abs(b); u=Math.abs(a)*Math.abs(b); System.out.println(s+" "+r+" "+d+" "+u); }</pre>	
Go	
<pre>package main import ("fmt" "math") func main() { fmt.Println("a, b=?") var a,b float64 fmt.Scanf("%v %v", &a, &b) var s=math.Abs(a)+math.Abs(b); var r = math.Abs(a) - math.Abs(b); var p = math.Abs(a) * math.Abs(b); var t = math.Abs(a) / math.Abs(b); fmt.Println("S=",s,"R=",r, "P=",p,"T=",t) }</pre>	

- 4) Даны два круга с общим центром и радиусами R_1 и R_2 ($R_1 > R_2$). Найти площади этих кругов S_1 и S_2 , а также площадь S_3 кольца, внешний радиус которого равен R_1 , а внутренний радиус равен R_2 :

$$S_1 = \pi \cdot (R_1)^2, \quad S_2 = \pi \cdot (R_2)^2, \quad S_3 = S_1 - S_2.$$

В качестве значения π использовать 3.14.

PascalABC.NET	C++
<pre> Var R1, R2, S1, S2, S3: Real; begin ReadLn(R1, R2); S1:= pi * Sqr(R1); S2:= pi * Sqr(R2); S3 := S1 - S2; WriteLn('S1=', S1, ' S2=',S2,' S3=', S3); end. </pre>	<pre> #define PI 3.14159265 #include <iostream> using namespace std; int main(){ double R1, R2, S1, S2, S3; cin >> R1 >> R2; S1 = PI * R1*R1; S2 = PI * R2*R2; S3 = S1 - S2; cout<<"S1 -> "<<S1<< "\nS2 -> " << S2 << "\nS3 ->" << S3 << endl; system("pause"); return 0; } </pre>
Python	C#
<pre> r1=float(input("r1:")) r2=float(input("r2:")) S1=3.14*r1*r1 S2=3.14*r2*r2 S=S1-S2 print(S) </pre>	<pre> public static void Main(string[] args){ double r1,r2, s1,s2,S; r1=Convert.ToDouble(Console.ReadLine()); r2=Convert.ToDouble(Console.ReadLine()); s1=Math.PI*r1*r1; s2=Math.PI*r2*r2; S=s1-s2; Console.WriteLine(S); Console.ReadKey(true); } </pre>
Java	GO
<pre> public static void main (String[] args) { Double S1,S2,S3,R1,R2; Scanner ld=new Scanner(System.in); R1=ld.nextDouble(); R2=ld.nextDouble(); S1=Math.PI*(R1*R1); S2=Math.PI*(R2*R2); S3=S1-S2; System.out.println(S1); System.out.println(S2); System.out.println(S3); } </pre>	<pre> package main import ("fmt" "math") func main() { fmt.Println("Insigth R1 and R2!") var R1,R2 float64 fmt.Scanf("%v %v", &R1, &R2) var S1 = 3.14 * math.Pow(R1,2); var S2 = 3.14 * math.Pow(R2,2); var S3 = S1 - S2; fmt.Println("S1=",S1,"\nS2=",S2, "\nS3=",S3) } </pre>

- 5) Найти расстояние между двумя точками с заданными координатами (x_1, y_1) и (x_2, y_2) на плоскости. Расстояние вычисляется по формуле

$$s = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}.$$

PascalABC.NET	C++
<pre> Var x1,y1,x2,y2,S: Real; begin WriteLn('x1, y1'); RealLn(x1, y1); WriteLn('x2, y2'); RealLn(x2, y2); </pre>	<pre> #define PI 3.14159265 #include <iostream> using namespace std; int main() { double x1, y1, x2, y2,s; cout << "x1, y1 -> "; cin >> x1 >> y1; </pre>

<pre>S=sqrt(sqr(x2-x1)+sqr(y2- y1)); WriteLn('S = ', S); end.</pre>	<pre>cout << "x2, y2 -> "; cin >> x2 >> y2; s=sqrt((y2 - y1) * (x2 - x1)+y2- y1)*(y2-y1)); cout<<"S="<<s<<endl ; system("pause"); return 0; }</pre>
<p style="text-align: center;">Python</p> <pre>from math import sqrt x1= float(input("x1:")) x2= float(input("x2:")) y1= float(input("y1:")) y2= float(input("y2:")) S=sqrt((x2- x1)**2+(y2-y1)**2) print(S)</pre>	<p style="text-align: center;">C#</p> <pre>public static void Main(string[] args){ double x1, x2, y1, y2, s; x1=Convert.ToDouble(Console.ReadLine()); y1=Convert.ToDouble(Console.ReadLine()); x2=Convert.ToDouble(Console.ReadLine()); y2=Convert.ToDouble(Console.ReadLine()); s=Math.Sqrt((x2 - x1) * (x2 - x1) + (y2 - y1) * (y2 - y1)); Console.WriteLine(s); Console.ReadKey(true); }</pre>
<p style="text-align: center;">Java</p> <pre>public static void main(String[] args) { double S,x1,x2,y1,y2; Scanner cp=new Scanner (System.in); x1 = cp.nextDouble(); x2= cp.nextDouble(); y1= cp.nextDouble(); y2=cp.nextDouble(); S=Math.sqrt((x2-x1)* (x2-x1)-(y2-y1)*(y2-y1)); System.out.println(S); }</pre>	<p style="text-align: center;">Go</p> <pre>package main import ("fmt" "math") func main() { fmt.Println("Insigth x1,x2 and y1,y2!") var x1,x2,y1,y2 float64 fmt.Scanf("%v %v %v %v", &x1, &x2, &y1, &y2) var S=math.Sqrt((x2-x1)*(x2-x1)-(y2- y1)*(y2-y1)); fmt.Println("S=",S) }</pre>

6) Поменять местами содержимое переменных *A* и *B* и вывести новые значения *A* и *B*.

<p style="text-align: center;">PascalABC.NET</p> <pre>var A, B, Temp: Real; begin WriteLn(' A, B'); RealLn(A, B); C := A; A := B; B := C; WriteLn('A =',A, ' B=',B); end.</pre>	<p style="text-align: center;">C++</p> <pre>#define PI 3.14159265 #include <iostream> using namespace std; int main(){ double A, B, C; cout << "A, B:"; cin >> A >> B; C = A; A = B; B = C; cout<<"A="<<A<<"\nB="<<B<<endl; system("pause"); return 0; }</pre>
--	--

Python	C#
<pre>a= int(input("a:")) b= int(input("b:")) (a, b)=(b,a); print(a, " ", b)</pre>	<pre>public static void Main(string[] args) { int a, b, c; a = Convert.ToInt(Console.ReadLine()); b = Convert.ToInt(Console.ReadLine()); c=a; a=b; b=c; Console.WriteLine(a+" "+b); Console.ReadKey(true); }</pre>
Java	Go
<pre>public static void main(String[] args) { int A, B,c; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); c=a; a=b; b=c; System.out.println(a+" "+b); }</pre>	<pre>package main import "fmt" func main() { var A,B float64 fmt.Scanf("%v %v", &A, &B) A,B=B,A; fmt.Println("A=",A,"\nB=",B) }</pre>

- 7) Дано число A . Вычислить A^8 , используя вспомогательную переменную и три операции умножения. Для этого последовательно находить A^2, A^4, A^8 . Вывести все найденные степени числа A .

PascalABC.NET	C++
<pre>Var A, Res: Real; begin ReadLn(A); Res := A * A; WriteLn('A^2 = ', Res); A := Res * Res; WriteLn('A^4 = ', A); Res := A * A; WriteLn('A^8 = ', Res); end.</pre>	<pre>#include <iostream> using namespace std; int main() { double A, Res; cout << "A -> "; cin >> A; Res = A * A; cout <<"A^2->" << Res; A = Res*Res; cout <<"A^4->" << A; Res = A*A; cout <<"A^8->" << Res; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input(«a:»)) b=a*a print(b) a=b*b b=a*a print(a, " ", b)</pre>	<pre>public static void Main(string[] args){ double a,c; a = Convert.ToInt32(Console.ReadLine()); c =a*a; Console.WriteLine(c); a=c*c; }</pre>

	<pre>c=a*a; Console.WriteLine(a+" "+c); Console.ReadKey(true); }</pre>
Java	Go
<pre>public static void main(String[] args) { double a,c; Scanner sc=new Scanner(System.in); a=sc.nextDouble(); c=a*a; System.out.print(c+" "); a=c*c; c=a*a; System.out.println(a+" "+c); }</pre>	<pre>package main import "fmt" func main() { var a,c float64 fmt.Scanf("%v", &a) c=a*a fmt.Println(c,"\t") a=c*c c=a*a fmt.Println(a,"\n",c) }</pre>

- 8) Дано значение угла π в градусах ($0 < \pi < 360$). Определить значение этого же угла в радианах, учитывая, что $180^\circ = \pi$ радианов. В качестве значения π использовать 3.14.

PascalABC.NET	C++
<pre>Var alfa:Real; Begin ReadLn(alfa); WriteLn(alfa*pi/180,' радиан'); End.</pre>	<pre>#define PI 3.14159265 #include <iostream> using namespace std; int main(){ double alfa; cout<<"ALFA->"; cin >> alfa; cout<<(alfa*PI)/180<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= float(input("a:")) x=(a*3.14)/180 print(x)</pre>	<pre>public static void Main(string[] args) { double x,a; a = Convert.ToSingle(Console.ReadLine()); x =(a* 3.14) / 180 ; Console.WriteLine(x); Console.ReadKey(true); }</pre>
Java	Go
<pre>public static void main(String[] args) { double a,x; Scanner sc=new Scanner(System.in); a=sc.nextDouble(); x=(Math.PI*a)/180; System.out.println(x); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a!") var a float64 fmt.Scanf("%v", &a) var x=(3.14*a)/180; fmt.Println("x=",x) }</pre>

- 9) Скорость лодки в стоячей воде V км/ч, скорость течения реки U км/ч ($U < V$). Время движения лодки по озеру T_1 ч, а по реке (против течения) — T_2 ч. Определить путь S , пройденный лодкой (путь = время · скорость). Учесть, что при движении против течения скорость лодки уменьшается на величину скорости течения.

PascalABC.NET	C++
<pre> Var V, U, T1, T2, S: Real; begin WriteLn('V,U,T1,T2:'); ReadLn(V, U, T1, T2); S := T1 * V + T2*(V-U); WriteLn('S= ', S); end. </pre>	<pre> #include <iostream> using namespace std; int main() { double V, U, T1, T2, S; cout<<"V,U,T1,T2->"; cin >> V >> U >> T1 >> T2; S = T1*V + T2*(V - U); cout<<S<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> u= float(input("u:")) v= float(input("v:")) t1= float(input("t1:")) t2= float(input("t2:")) S=t1*v+(v-u)*t2; print(S) </pre>	<pre> public static void Main(string[] args) { double v, u, t1, t2, s; v=Convert.ToInt32(Console.ReadLine()); u=Convert.ToInt32(Console.ReadLine()); t1=Convert.ToInt32(Console.ReadLine()); t2=Convert.ToInt32(Console.ReadLine()); s = v * t1 + (v - u) * t2; Console.WriteLine(s); Console.ReadKey(true); } </pre>
Java	Go
<pre> public static void main (String[] args) { double u,v,t1,t2,S; Scanner sc=new Scanner (System.in); v=sc.nextDouble(); u=sc.nextDouble(); t1=sc.nextDouble(); t2=sc.nextDouble();; S= t1*v+(v-u)*t2; System.out.println(S); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("u,v,t1,t2!") var u,v,t1,t2 float64 fmt.Scanf("%v %v %v %v", &v, &u,&t1,&t2) var S=t1*v+(v-u)*t2 fmt.Println("S=",S) } </pre>

- 10) Найти корни квадратного уравнения $A \cdot x^2 + B \cdot x + C = 0$, заданного своими коэффициентами A , B , C (коэффициент A не равен 0), если известно, что дискриминант уравнения положителен. Вывести вначале меньший, а затем больший из найденных корней. Корни квадратного уравнения находятся по формуле $x_{1,2} = \frac{-B \pm \sqrt{D}}{2 \cdot A}$, где D — дискриминант, равный $D = B^2 - 4 \cdot A \cdot C$.

PascalABC.NET	C++
<pre> Var a, b, c, x1, x2: real; begin writeln('A, B,c:'); readln(a, b, c); d:=sqr(b) - 4 * a * c; x1:=(-b-sqrt(d))/(2*a); x2:=(-b+sqrt(d))/(2*a); writeln('x1 = ', x1, ' x2 = ', x2); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ double a, b, c, x1, x2,d; cout << "enter a, b and c -> "; cin>>a>>b>>c; d = b*b - 4*a*c; x1 = (-b - sqrt(d))/(2*a); x2 = (-b + sqrt(d))/(2*a); cout<<"x1="<<x1<<"\nx2="<<x2<<endl;; system("pause"); return 0; } </pre>
Python	C#
<pre> from math import sqrt a= float(input("a:")) b= float(input("b:")) c= float(input("c:")) d=b*b-4*a*c x1=(-b+sqrt(d))/2*a x2=(-b-sqrt(d))/2*a print(x1," ",x2) </pre>	<pre> public static void Main(string[] args){ double a, b, c, x1, x2, d; a = Convert.ToDouble(Console.ReadLine()); b = Convert.ToDouble(Console.ReadLine()); c = Convert.ToDouble(Console.ReadLine()); d = b * b - 4 * a * c; x1 = (-b * Math.Sqrt(d)) / (2 * a); x2 = (-b - Math.Sqrt(d)) / (2 * a); Console.WriteLine(x1 + " " + x2); Console.ReadKey(true); } </pre>
Java	Go
<pre> public static void main(String[] args){ double A, B,c,d,x1,x2; Scanner cp=new Scanner(System.in); a=cp.nextDouble(); b=cp.nextDouble(); c=cp.nextDouble(); d=b*b-4*a*c; x1=(-b+Math.sqrt(d))/2*a; x2=(-b+Math.sqrt(d))/2*a; System.out.println(x1+" "+x2); } </pre>	<pre> package main import ("fmt" "math") func main() { fmt.Println("a,b,c") var a,b,c float64 fmt.Sprintf("%v %v %v",&a,&b,&c) var d = b*b - 4*a*c; var x1=(-b-math.Sqrt(d)) / (2*a); var x2=(-b+math.Sqrt(d))/(2*a); fmt.Println("x1=",x1,"\nx2=",x2) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextDouble() var b=reader.nextDouble() var c=reader.nextDouble() val d=b*b-4*a*c var x1=(-b+Math.sqrt(d))/2*a val x2=(-b-Math.sqrt(d))/2*a println("x1=\$x1 x2=\$x2") } </pre>	

Задачи для самостоятельной работы

Все входные и выходные данные в заданиях этой группы являются вещественными числами.

- 1) Дана сторона квадрата a . Найти его площадь $S = a^2$.
- 2) Даны стороны прямоугольника a и b . Найти его площадь $S = a \cdot b$ и периметр $P = 2 \cdot (a + b)$.
- 3) Дана длина ребра куба a . Найти объем куба $V = a^3$ и площадь его поверхности $S = 6 \cdot a^2$.
- 4) Даны длины ребер a, b, c прямоугольного параллелепипеда. Найти его объем $V = a \cdot b \cdot c$ и площадь поверхности $S = 2 \cdot (a \cdot b + b \cdot c + a \cdot c)$.
- 5) Найти длину окружности L и площадь круга S заданного радиуса R :

$$L = 2 \cdot \pi \cdot R, \quad S = \pi \cdot R^2.$$

В качестве значения π использовать 3.14.

- 6) Даны два числа a и b . Найти их *среднее арифметическое*: $s = \frac{a+b}{2}$.
- 7) Даны два неотрицательных числа a и b . Найти их *среднее геометрическое*, то есть квадратный корень из их произведения: $c = \sqrt{a \cdot b}$.
- 8) Даны два ненулевых числа. Найти сумму, разность, произведение и частное их квадратов.
- 9) Даны катеты прямоугольного треугольника a и b . Найти его гипотенузу c и периметр P :

$$c = \sqrt{a^2 + b^2}, \quad P = a + b + c.$$

- 10) Дана длина L окружности. Найти ее радиус R и площадь S круга, ограниченного этой окружностью, учитывая, что $L = 2 \cdot \pi \cdot R$, $S = \pi \cdot R^2$. В качестве значения π использовать 3.14.
- 11) Дана площадь S круга. Найти его диаметр D и длину L окружности, ограничивающей этот круг, учитывая, что $L = \pi \cdot D$, $S = \pi \cdot D^2/4$. В качестве значения π использовать 3.14.
- 12) Найти расстояние между двумя точками с заданными координатами x_1 и x_2 на числовой оси: $s = |x_2 - x_1|$.
- 13) Даны три точки A, B, C на числовой оси. Найти длины отрезков AC и BC и их сумму.
- 14) Даны три точки A, B, C на числовой оси. Точка C расположена между точками A и B . Найти произведение длин отрезков AC и BC .
- 15) Даны координаты двух противоположных вершин прямоугольника: (x_1, y_1) , (x_2, y_2) . Стороны прямоугольника параллельны осям координат. Найти периметр и площадь данного прямоугольника.
- 16) Даны координаты трех вершин треугольника: (x_1, y_1) , (x_2, y_2) , (x_3, y_3) . Найти его периметр и площадь, используя формулу для расстояния между двумя точками на плоскости. Для нахождения площади треугольника со сторонами a, b, c использовать *формулу Герона*:

$$S = \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)},$$

где $p = (a + b + c)/2$ — *полупериметр*.

- 17) Даны переменные A, B, C . Изменить их значения, переместив содержимое A в B , B — в C , C — в A , и вывести новые значения переменных A, B, C .

- 18) Даны переменные A, B, C . Изменить их значения, переместив содержимое A в C , C — в B , B — в A , и вывести новые значения переменных A, B, C .
- 19) Найти значение функции $y = 3 \cdot x^6 - 6 \cdot x^2 - 7$ при данном значении x .
- 20) Найти значение функции $y = 4(x - 3)^6 - 7(x - 3)^3 + 2$ при данном значении x .
- 21) Дано число A . Вычислить A^{15} , используя две вспомогательные переменные и пять операций умножения. Для этого последовательно находить $A^2, A^3, A^5, A^{10}, A^{15}$. Вывести все найденные степени числа A .
- 22) Дано значение угла π в радианах ($0 < \pi < 2 \cdot \pi$). Определить значение этого же угла в градусах, учитывая, что $180^\circ = \pi$ радианов. В качестве значения π использовать 3.14.
- 23) Дано значение температуры T в градусах Фаренгейта. Определить значение этой же температуры в градусах Цельсия. Температура по Цельсию T_C и температура по Фаренгейту T_F связаны следующим соотношением:
- $$T_C = (T_F - 32) \cdot 5/9.$$
- 24) Дано значение температуры T в градусах Цельсия. Определить значение этой же температуры в градусах Фаренгейта. Температура по Цельсию T_C и температура по Фаренгейту T_F связаны следующим соотношением:
- $$T_C = (T_F - 32) \cdot 5/9.$$
- 25) Известно, что X кг шоколадных конфет стоит A рублей, а Y кг ирисок стоит B рублей. Определить, сколько стоит 1 кг шоколадных конфет, 1 кг ирисок, а также во сколько раз шоколадные конфеты дороже ирисок.
- 26) Скорость первого автомобиля V_1 км/ч, второго — V_2 км/ч, расстояние между ними S км. Определить расстояние между ними через T часов, если автомобили удаляются друг от друга. Данное расстояние равно сумме начального расстояния и общего пути, проделанного автомобилями; общий путь = время \cdot суммарная скорость.
- 27) Скорость первого автомобиля V_1 км/ч, второго — V_2 км/ч, расстояние между ними S км. Определить расстояние между ними через T часов, если автомобили первоначально движутся навстречу друг другу. Данное расстояние равно модулю разности начального расстояния и общего пути, проделанного автомобилями; общий путь = время \cdot суммарная скорость.
- 28) Решить линейное уравнение $A \cdot x + B = 0$, заданное своими коэффициентами A и B (коэффициент A не равен 0).
- 29) Найти решение системы линейных уравнений вида
- $$\begin{cases} A_1 \cdot x + B_1 \cdot y = C_1, \\ A_2 \cdot x + B_2 \cdot y = C_2. \end{cases}$$
- заданной своими коэффициентами $A_1, B_1, C_1, A_2, B_2, C_2$, если известно, что данная система имеет единственное решение. Воспользоваться формулами
- $$\begin{cases} x = \frac{C_1 \cdot B_2 - C_2 \cdot B_1}{D} \\ y = \frac{A_1 \cdot C_2 - A_2 \cdot C_1}{D}. \end{cases}$$
- где $D = A_1 \cdot B_2 - A_2 \cdot B_1$.
- 30) Составить программу:
- а) вычисления значения функции $y = 10x^2 + 4x - 5$ при любом значении x ;

- б) вычисления значения функции $x = 4a^2 - 12a + 3$ при любом значении a ;
- г) вычисления значения функции $y = \frac{a^2+8}{\sqrt{a^2+2}}$ при любом значении a ;
- 31) Известны объем и масса тела. Определить плотность материала этого тела.
- 32) Известны количество жителей в государстве и площадь его территории.
Определить плотность населения в этом государстве.
- 33) Треугольник задан координатами своих вершин. Найти периметр и площадь треугольника.
- 34) Выпуклый четырехугольник задан координатами своих вершин. Найти площадь этого четырехугольника как сумму площадей треугольников.
- 35) Известна стоимость 1 кг конфет, печенья и яблок. Найти стоимость всей покупки, если купили x кг конфет, y кг печенья и z кг яблок.
- 36) Известна стоимость монитора, системного блока, клавиатуры и мыши.
Сколько будут стоить 3 компьютера из этих элементов? N компьютеров?
- 37) Возраст Тани — X лет, а возраст Мити — Y лет. Найти их средний возраст, а также определить, на сколько отличается возраст каждого ребенка от среднего значения.
- 38) Даны две целые переменные a, b . Составить фрагмент программы, после исполнения, которого значения переменных поменялись бы местами (не используя новую переменную).
- 39) Найти суммарный объем n цилиндров высоты h , если радиусы их оснований образуют последовательность $3, 6, 9, \dots, 3n$. Найти среднюю длину радиуса.
- 40) Вероятность того, что новая шариковая ручка пишет плохо или вовсе не пишет, равна N ($0 < N < 1$). Покупатель, не глядя, берёт одну шариковую ручку из коробки. Найдите вероятность того, что эта ручка пишет хорошо.
- 41) В случайном эксперименте симметричную монету бросают N раз. Найдите вероятность того, что орёл выпадет хотя бы один раз.
- 42) В урне находится W белых, R красных и B черных шаров. Наугад извлекается 1 шар. Найти вероятность того, что он будет красным.
- 43) На координатной плоскости заданы точки $A(x_1; y_1)$, $B(x_2; y_2)$ и $C(x_3; y_3)$.
Найдите тангенс угла между векторами AB и AC .
- 44) Дана площадь прямоугольника. Найдите его стороны, для которого периметр будет наибольшим.
- 45) Две точки на плоскости $A(x_1; y_1)$ и $B(x_2; y_2)$ заданы своими координатами.
Найти уравнение прямой, проходящей через эти точки.
- 46) Диаметр колеса автомобиля 80 см. Колесо потребует замены через 200 000 оборотов. Определить, доедет ли колесо от города A до города B , если расстояние между ними N километров.
- 47) Даны два вектора \vec{A} и \vec{B} . Найти сумму векторов \vec{A} и \vec{B} .
- 48) Даны два вектора \vec{A} и \vec{B} . Найти вектора $\vec{C} = 8\vec{A}$ и $\vec{D} = -6\vec{A} - 10\vec{B}$.
- 49) Даны два вектора \vec{A} и \vec{B} . Найти скалярное произведение векторов \vec{A} и \vec{B} .
- 50) Даны два вектора \vec{A} и \vec{B} . Найти угол между векторами \vec{A} и \vec{B} .

2. Целые числа: группа Integer

Все входные и выходные данные в заданиях этой группы являются целыми числами. Все числа, для которых указано количество цифр (двухзначное число, трехзначное число и т. д.), считаются положительными.

- 1) Дано расстояние L в сантиметрах. Используя операцию деления нацело, найти количество полных метров в нем (1 метр = 100 см).

PascalABC.NET	C++
<pre>var L, M: integer; begin read(L); M := L div 100; write(M); end.</pre>	<pre>#include <iostream> using namespace std; int main() { int L,M; cin>>L; M=L/100; cout<<"M="<<M<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>N= int(input("n:")) M=int(L/100); print(M);</pre>	<pre>static void Main(string[] args) { int L,M; L= Convert.ToInt32(Console.ReadLine()); M = L/ 1000; Console.Write(M); Console.ReadKey(); }</pre>
Kotlin	
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var L=reader.nextInt() var m=L/100 println(m) }</pre>	
VB.NET	
<pre>Sub Main() Dim L, m As Integer L = Convert.ToInt32(Console.ReadLine()) m = L \ 100 Console.WriteLine(m) Console.ReadKey() End Sub</pre>	

- 2) Даны целые положительные числа A и B ($A > B$). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Используя операцию деления нацело, найти количество отрезков B , размещенных на отрезке A .

PascalABC.NET	C++
<pre> Var A, B, Res: Integer; begin Write('A: '); ReadLn(A); Write('B: '); ReadLn(B); Res := A div B; WriteLn(Res); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int A, B, Res; cin>>A>>B; Res=A/B; cout<< Res<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("a: ")) b= int(input("b: ")) c=a//b print(c) </pre>	<pre> static void Main(string[] args){ int a, b, y; a= Convert.ToInt32(Console. ReadLine()); b = Convert.ToInt32(Console. ReadLine()); y = a/ b; Console.WriteLine(y); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { int B,A,L; Scanner sc=new Scanner (System.in); A=sc.nextInt(); B=sc.nextInt(); L=A/B; System.out.println(L); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("a,b") var a,b int fmt.Scanf("%d %d", &a,&b) var L=a/b fmt.Println("L=",L) } </pre>

3) Дано двузначное число. Вывести вначале его левую цифру (десятки), а затем — его правую цифру (единицы). Для нахождения десятков использовать операцию деления нацело, для нахождения единиц — операцию взятия остатка от деления.

PascalABC.NET	C++
<pre> Var a, r, L: integer; begin WriteLn('a'); ReadLn(a); r := a div 10; L := a mod 10; WriteLn('r=', r); WriteLn('L=', L); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int A, B,c; cout<<"n=: "; cin>>a; b = a / 10; c = a % 10; cout<<b<<" " <<c; system("pause"); return 0; } </pre>

Python	C#
<pre>n= int(input("n:")) a=n//10 b=n%10 print(a, ' ',b)</pre>	<pre>static void Main(string[] args){ int L, m,c; L = Convert.ToInt32(Console.ReadLine()); m = L / 10; c = L % 10; Console.WriteLine(m+" "+c); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ int A, B,N; Scanner sc=new Scanner(System.in); N=sc.nextInt(); a=N/10; b=N%10; System.out.println(a+" "+b); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a->!") var n int fmt.Scanf("%d", &n) var a=n/10; var b=n%10; fmt.Println("a=",a,"\nb=",b) }</pre>
Kotlin	
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var n=reader.nextInt() var a=n/10 var b=n%10 println("a=\$a b=\$b") }</pre>	
VB.NET	
<pre>Sub Main() Dim n, a, b As Integer n = Convert.ToInt32(Console.ReadLine()) a = n \ 10 b = n Mod 10 Console.WriteLine("{0} {1}", a, b) Console.ReadKey() End Sub</pre>	

4) Дано трехзначное число. Используя одну операцию деления нацело, вывести первую цифру данного числа (сотни).

PascalABC.NET	C++
<pre>Var a, r: integer; begin Writeln('a'); readln(a); r := a div 100; Writeln('r=', r); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int a; cout<<"a: "; cin>>a; a = a / 100; cout<<a<<endl; system("pause"); return 0; }</pre>

Python	C#
<pre>n= int(input("n:")) a=n//100 print(a)</pre>	<pre>static void Main(string[] args){ int l, m; l=Convert.ToInt32(Console.ReadLine()); m = l / 100; Console.WriteLine(m); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { int a,N; Scanner sc=new Scanner(System.in); N=sc.nextInt(); a=N/100; System.out.println(a); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a->") var n int fmt.Scanf("%d", &n) var a=n/100; fmt.Println("Сади=",a) }</pre>

5) Дано трехзначное число. Вывести число, полученное при прочтении исходного числа справа налево.

PascalABC.NET	C++
<pre>Var x, a, b, c, p: integer; begin Writeln('a'); readln(x); a := x div 100; b:=(x div 10) mod 10; c := x mod 10; p:=c*100+b*10+a; Writeln('p =', p); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int A, B,c,d,f; cout<<"n:"; cin>>a; b = a / 100; c = a % 100; d = c % 10; f = c / 10; cout<<d<<" "<<f<<" "<<b<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>n= int(input("n: ")) a=n//100 b=(n%100)//10 c=n%10 S=c*100+b*10+a print(S)</pre>	<pre>static void Main(string[] args){ int a, b,c,x,m,n ; n = Convert.ToInt32(Console.ReadLine()); a = n / 100; x = n % 100; b = x / 10; c = x % 10; m = 100 * c + 10 * b + a; Console.WriteLine(m); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { int A, B,c,N,M; Scanner sc=new Scanner(System.in); N=sc.nextInt();</pre>	<pre>package main import "fmt" func main() { fmt.Println("n->") var n int</pre>

<pre> a=N/100; b=(N%100)/10; c=(N%10)%10; M=100*c+10*b+a; System.out.println(M); } </pre>	<pre> fmt.Scanf("%d", &n) var a=n/100; var b=(n%100)/10; var c=(n%10)%10; var M=100*c+10*b+a; fmt.Println("M=",M) } </pre>
---	--

6) С начала суток прошло N секунд (N — целое). Найти количество полных минут, прошедших с начала суток.

PascalABC.NET	C++
<pre> Var s, m: integer; begin readln(s); m := s div 60; writeln('m =', m) end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int t;cin>>t; t = t / 60; cout<<"m="<<t; system("pause"); return 0; } </pre>
Python	C#
<pre> s= int(input("s:")) m=s//60 print(m) </pre>	<pre> static void Main(string[] args){ int m,n ; n = Convert.ToInt32(Console.ReadLine()); m = n / 60; Console.WriteLine(m); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args){ int Nc,M; Scanner sc=new Scanner(System.in); Nc=sc.nextInt(); M=Nc/60; System.out.println(M); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("n>999") var ns,M int fmt.Scanf("%d", &ns) M=ns/60; fmt.Println("M=",M) } </pre>

7) Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, ..., 6 — суббота. Дано целое число K , лежащее в диапазоне 1–365. Определить номер дня недели для K -го дня года, если известно, что в этом году 1 января было понедельником.

PascalABC.NET	C++
<pre> Var x, d: integer; begin readln(x); d := x mod 7; writeln(d); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int d; cin>>d; d = d % 7; cout<<d<<endl; system("pause"); return 0; } </pre>

Python	C#
<pre>s= int(input("s:")) m=s%7 print(m)</pre>	<pre>static void Main(string[] args){ int y,k ; k=Convert.ToInt32(Console.ReadLine()); y = k % 7; Console.WriteLine(y); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ int k,L; Scanner lk=new Scanner(System.in); k=lk.nextInt(); L=k%7; System.out.println(L); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("k->") var k,L int fmt.Scanf("%d", &k) L=k%7; fmt.Println(L) }</pre>

8) Даны целые положительные числа A, B, C . На прямоугольнике размера $A \neq B$ размещено максимально возможное количество квадратов со стороной C (без наложений). Найти количество квадратов, размещенных на прямоугольнике, а также площадь незанятой части прямоугольника.

PascalABC.NET	C++
<pre>Var a, b, c, s, p: integer; Resulr: real; begin readln(a, b, c); p:=(a div c)*(b div c); s := b * (a mod c) + (b mod c)*(a-a mod c); writeln(s, ' ', p); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int A, B,c,p,s; cin>>a>>b>>c; p = (a/c) * (b/c); s = b*(a % c) + (b % c)*(a-a%c); cout<<p<<" "<<s<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input(«a:»)) b= int(input(«b:»)) c= int(input(«c:»)) p = (a//c) * (b//c); s = b*(a % c) + (b % c)*(a-a%c); print(p, ' ',s)</pre>	<pre>static void Main(string[] args){ int A, B,c,s,p; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); c = Convert.ToInt32(Console.ReadLine()); p = (a/c) * (b/c); s = b*(a % c) + (b % c)*(a-a%c); Console.WriteLine(p + " " + s); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ int A, B,c,p,s; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); c=sc.nextInt(); p = (a/c) * (b/c); s = b*(a % c) + (b % c)*(a-a%c); System.out.println(p+" "+s); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a,b,c->") var a,b,c,s,p int fmt.Scanf("%d %d %d",&a,&b,&c) p = (a/c) * (b/c); s=b*(a % c)+(b%c)*(a-a%c); fmt.Println("P=",p,"\nS=",s) }</pre>

Задачи для самостоятельной работы

Все входные и выходные данные в заданиях этой группы являются целыми числами. Все числа, для которых указано количество цифр (двухзначное число, трехзначное число и т. д.), считаются положительными.

- 1) Дана масса M в килограммах. Используя операцию деления нацело, найти количество полных тонн в ней (1 тонна = 1000 кг).
- 2) Дан размер файла в байтах. Используя операцию деления нацело, найти количество полных килобайтов, которые занимает данный файл (1 килобайт = 1024 байта).
- 3) Даны целые положительные числа A и B ($A > B$). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Используя операцию взятия остатка от деления нацело, найти длину незанятой части отрезка A .
- 4) Дано двухзначное число. Найти сумму и произведение его цифр.
- 5) Дано двухзначное число. Вывести число, полученное при перестановке цифр исходного числа.
- 6) Дано трехзначное число. Используя одну операцию деления нацело, вывести первую цифру данного числа (сотни).
- 7) Дано трехзначное число. Вывести вначале его последнюю цифру (единицы), а затем — его среднюю цифру (десятки).
- 8) Дано трехзначное число. Найти сумму и произведение его цифр.
- 9) Дано трехзначное число. В нем зачеркнули первую слева цифру и приписали ее справа. Вывести полученное число.
- 10) Дано трехзначное число. В нем зачеркнули первую справа цифру и приписали ее слева. Вывести полученное число.
- 11) Дано трехзначное число. Вывести число, полученное при перестановке цифр сотен и десятков исходного числа (например, 123 перейдет в 213).
- 12) Дано трехзначное число. Вывести число, полученное при перестановке цифр десятков и единиц исходного числа (например, 123 перейдет в 132).
- 13) Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду сотен в записи этого числа.
- 14) Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду тысяч в записи этого числа.
- 15) С начала суток прошло N секунд (N — целое). Найти количество полных часов, прошедших с начала суток.
- 16) С начала суток прошло N секунд (N — целое). Найти количество секунд, прошедших с начала последней минуты.
- 17) С начала суток прошло N секунд (N — целое). Найти количество секунд, прошедших с начала последнего часа.
- 18) С начала суток прошло N секунд (N — целое). Найти количество полных минут, прошедших с начала последнего часа.
- 19) Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, ..., 6 — суббота. Дано целое число K , лежащее в диапазоне 1–365. Определить номер дня недели для K -го дня года, если известно, что в этом году 1 января было четвергом.

- 20) Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, ..., 6 — суббота, 7 — воскресенье. Дано целое число K , лежащее в диапазоне 1–365. Определить номер дня недели для K -го дня года, если известно, что в этом году 1 января было вторником.
- 21) Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, ..., 6 — суббота, 7 — воскресенье. Дано целое число K , лежащее в диапазоне 1–365. Определить номер дня недели для K -го дня года, если известно, что в этом году 1 января было субботой.
- 22) Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, ..., 6 — суббота, 7 — воскресенье. Дано целое число K , лежащее в диапазоне 1–365, и целое число N , лежащее в диапазоне 1–7. Определить номер дня недели для K -го дня года, если известно, что в этом году 1 января было днем недели с номером N .
- 23) Дан номер некоторого года (целое положительное число). Определить соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.
- 24) С некоторого момента прошло N дней. Сколько полных недель прошло за этот период?
- 25) С некоторого момента прошло N месяцев. Сколько полных лет прошло за этот период?
- 26) С некоторого момента прошло N месяцев. Сколько полных столетий прошло за этот период?
- 27) Дано натуральное число N . Найти число единиц в нем.
- 28) Дано натуральное число N . Найти число десятков в нем.
- 29) Из трехзначного числа x вычли его последнюю цифру. Когда результат разделили на 10, а к частному слева приписали последнюю цифру числа x , то получилось число N . Найти число x .
- 30) В трехзначном числе x зачеркнули первую цифру. Когда оставшееся число умножили на 10, а произведение сложили с первой цифрой числа x , то получилось число N . Найти число x .
- 31) В трехзначном числе x зачеркнули его вторую цифру. Когда к образованному при этом двузначному числу слева приписали вторую цифру числа x , то получилось число N . Найти число x .
- 32) В трехзначном числе x зачеркнули его вторую цифру. Когда к образованному при этом двузначному числу справа приписали вторую цифру числа x , то получилось число n . По заданному n найти число x (значение n вводится с клавиатуры, $100 \leq n \leq 999$).
- 33) Прямоугольник $N \times M$ разбит на клетки 1×1 . Требуется закрасить 4 клетки, образующие квадрат 2×2 . Сколькими разными способами можно выбрать такой квадрат?
- 34) Прямоугольник $N \times M$ разбит на клетки 1×1 . Требуется закрасить K^2 клетки, образующие квадрат $K \times K$. Сколькими разными способами можно выбрать такой квадрат?

- 35) Прямоугольник $N \times M$ разбит на клетки 1×1 . Требуется закрасить $K \times L$ клетки, образующие прямоугольник $K \times L$. Сколькими разными способами можно выбрать такой прямоугольник?
- 36) С начала 1990 года по некоторый день прошло N месяцев и 2 дня. Присвоить целочисленной величине x значение 1, 2, ..., 11 или 12 в зависимости от того, каким месяцем (январем, февралем и т. п.) является месяц этого дня. Например, при 3 и значение x равно 4.
- 37) Даны целые числа h, m, s ($0 < h \leq 23, 0 \leq m \leq 59, 0 \leq s \leq 59$), указывающие момент времени: " h часов, m минут, s секунд". Определить угол (в градусах) между положением часовой стрелки в начале суток и в указанный момент времени.
- 38) С начала суток часовая стрелка повернулась на y градусов ($0 \leq y < 360$, y — вещественное число). Определить число полных часов и число полных минут, прошедших с начала суток.
- 39) Улитка за день заползает вверх по дереву на N метров, а за ночь сползает на M ($M < N$). Высота дерева K метров. За сколько дней улитка впервые достигнет вершины дерева?
- 40) В трехзначном числе зачеркнули старшую цифру. Когда полученное число умножили на 7, то получили исходное число. Найти это число.
- 41) Даны цифры двух целых чисел: двузначного $a_2 a_1$ и однозначного b , где a_1 — число единиц, a_2 — число десятков. Получить цифры числа, равного сумме заданных чисел (известно, что это число двузначное). Слагаемое — двузначное число и число-результат не определять; условный оператор не использовать.
- 42) Даны цифры двух двузначных чисел, записываемых в виде $a_2 a_1$ и однозначного $b_2 b_1$, где a_1 и b_1 — число единиц, a_2 и b_2 — число десятков. Получить цифры числа, равного сумме заданных чисел (известно, что это число двузначное). Слагаемое — двузначное число и число-результат не определять; условный оператор не использовать.
- 43) Даны два целых числа a и b . Если a делится на b или b делится на a , то вывести 1, иначе — любое другое число. Условные операторы и операторы цикла не использовать.
- 44) Даны целое число k ($1 \leq k \leq 180$) и последовательность цифр 10111213...9899, в которой выписаны подряд все двузначные числа. Определить:
- номер пары цифр, в которую входит k - я цифра;
 - двузначное число, образованное парой цифр, в которую входит k - я цифра;
 - k - ю цифру, если известно, что:
 - k — четное число;
 - k — нечетное число.

Величины строкового типа не использовать.

3. Логические выражения: группа Boolean

Во всех заданиях данной группы требуется вывести логическое значение True, если приведенное высказывание для предложенных исходных данных является истинным, и значение False в противном случае. Все числа, для которых указано количество цифр (двухзначное число, трехзначное число и т. д.), считаются целыми положительными.

- 1) Дано целое число A. Проверить истинность высказывания: «Число A является положительным».

PascalABC.NET	C++
<pre>var A: Integer; begin ReadLn(A); WriteLn(A > 0); end.</pre>	<pre>#include <iostream> using namespace std; int main() { int A; cin >> A; cout << (A > 0); system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input("a: ")) y=a>0 print(y)</pre>	<pre>static void Main(string[] args){ int a; bool y; a = Convert.ToInt32(Console.ReadLine()); y = a > 0; Console.WriteLine(y); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ boolean y; int A ; Scanner sc=new Scanner(System.in); A=sc.nextInt(); y=A>0; System.out.println(y); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a=?") var a int var y bool fmt.Sprintf("%d", &a) y=a>0 fmt.Println("y=", y) }</pre>
Kotlin	
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextInt() var y=a>0 println("y=\$y") }</pre>	
VB.NET	
<pre>Sub Main() Dim a As Integer Dim y As Boolean a = Convert.ToInt32(Console.ReadLine())</pre>	

```

y = a > 0
Console.WriteLine(y)
Console.ReadKey()
End Sub

```

2) Даны два целых числа: A , B . Проверить истинность высказывания: «Справедливы неравенства $A > 0$ или $B < -2$ ».

PascalABC.NET	C++
<pre> Var A, B: Integer; y: Boolean; begin ReadLn(A, B); y:=(A>2) And (B<=3); WriteLn(y); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int A, B; cin >> A >> B; cout << ((A > 2) && (B <= 3)); system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("a: ")) b= int(input("b: ")) y=(a>2) and (b<=3) print(y) </pre>	<pre> static void Main(string[] args){ bool y; int a, b; a= Convert.ToInt32(Console.ReadLine()); b= Convert.ToInt32(Console.ReadLine()); y = (a > 2) && (b <= 3); Console.WriteLine(y); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args){ boolean y; int A, B ; Scanner sc=new Scanner(System.in); A=sc.nextInt(); B=sc.nextInt(); y=(A>2)&&(B<=3); System.out.println(y); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("a=?") var a,b int var y bool fmt.Scanf("%d %d",&a,&b) y=(a>2) && (b<=3) fmt.Println("y=", y) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextInt() var b=reader.nextInt() var y=(a>2) && (b<=3) println("y=\$y") } </pre>	

3) Даны два целых числа: A , B . Проверить истинность высказывания: «Каждое из чисел A и B нечетное».

PascalABC.NET	C++
<pre> Var A, B: Integer; begin ReadLn(A, B); WriteLn(Odd(A) And Odd(B)); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int A, B; cin >> A >> B; bool y=((A % 2 != 0)&&(B % 2 != 0)); } </pre>

	<pre> cout << y; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input(«a:»)) b=int(input("b: ")) y=(a%2)*(b%2)==1 print(y) </pre>	<pre> static void Main(string[] args) { int a, b; bool y; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); y = ((a % 2 == 0) && ((a % 2 == 0))); Console.WriteLine(y); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { boolean y; int A, B ; Scanner sc=new Scanner(System.in); A=sc.nextInt(); B=sc.nextInt(); y=((A%2==1)&&(B%2==1)); System.out.println(y); } </pre>	<pre> package main import ("fmt") func main() { fmt.Println("a, b") var a,b int var y bool fmt.Scanf("%d %d",&a,&b) y=((a%2==1)&&(b%2==1)); fmt.Println("Y=",y) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextInt() var b=reader.nextInt() var y=(a%2==0) && (b%2==0) println("y=\$y") } </pre>	

4) Даны два целых числа: A, B . Проверить истинность высказывания: «Ровно одно из чисел A и B нечетное».

PascalABC.NET	C++
<pre> Var A, B: Integer; begin ReadLn(A, B); WriteLn(Odd(A) Xor Odd(B)); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a, b; bool y; cin >> a >> b; y=((b%2==1) &&(a%2==0)) ((a%2==0)&&(b%2==1)); cout << y<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("a: ")) b= int(input("b: ")) y=(a%2)+(b%2)==1 print(y) </pre>	<pre> static void Main(string[] args){ int a, b; bool y; a=Convert.ToInt32(Console.ReadLine()); </pre>

	<pre> b=Convert.ToInt32(Console.ReadLine()); y = ((b % 2 == 1) &&(a % 2 == 0)) ((a%2==0)&&(b%2==1)); Console.WriteLine(y); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args) { boolean y; int A, B; Scanner sc=new Scanner (System.in); A=sc.nextInt(); B=sc.nextInt(); y=((A%2==1)&&(B%2==0)) ((A%2==0)&& (B%2==1)); System.out.println(y); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("A, B") var A,B int var y bool fmt.Scanf("%d %d",&A,&B) y=((A%2==1)&&(B%2==0)) ((A%2==0)&&(B%2==1)); fmt.Println("Y=",y) } </pre>

5) Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара совпадающих».

PascalABC.NET	C++
<pre> Var A, B, C: Integer; y: Boolean; begin ReadLn(A, B, C); y:=(A=B) Or (B=C) Or (A=C); WriteLn(y); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int A, B, C; cin >> A >> B >> C; bool y=((A == B) (B == C) (A == C)); cout <<y<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("a: ")) b = int(input("b:")) c = int(input("c:")) y=(a==b) or (a==c) or (b==c) print(y) </pre>	<pre> static void Main(string[] args) { int a, b, c; bool y; a= Convert.ToInt32(Console.ReadLine()); b= Convert.ToInt32(Console.ReadLine()); c= Convert.ToInt32(Console.ReadLine()); y=(a==b) a==c (b == c); Console.WriteLine(y); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args){ boolean y; int A, B,c; Scanner sc=new Scanner(System.in); a=sc.nextInt(); </pre>	<pre> package main import "fmt" func main() { var a,b,c int var y bool </pre>

<pre>b=sc.nextInt(); c=sc.nextInt(); y=(a==b) (b==c) (c==a); System.out.println(y); }</pre>	<pre>fmt.Scanf("%d %d %d", &a,&b,&c) y=(a==b) (b==c) (c==a); fmt.Println("Y=",y) }</pre>
---	--

6) Даны числа x, y . Проверить истинность высказывания: «Точка с координатами (x, y) лежит во второй координатной четверти».

PascalABC.NET	C++
<pre>var x, y: Real; z: Boolean; begin ReadLn(x, y); z:=(x<=0) And (y>=0); WriteLn(z); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ double x, y; cin >> x >> y; cout<<((x <= 0) && (y >= 0)); system("pause"); return 0; }</pre>
Python	C#
<pre>x= int(input(«x:»)) y = float(input(":")) z=(x<0) and (y>0) print(z)</pre>	<pre>static void Main(string[] args){ bool L; int x,y; x = Convert.ToInt32(Console.ReadLine()); y = Convert.ToInt32(Console.ReadLine()); L = (x < 0) && (y > 0); Console.WriteLine(L); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ boolean L; Double x,y; Scanner sc=new Scanner (System.in); x=sc.nextDouble(); y=sc.nextDouble(); L=(x<0)&&(y>0); System.out.println(L); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("x,y->") var x,y int var L bool fmt.Scanf("%d %d",&x,&y) L=(x<0)&&(y>0); fmt.Println("L=",L)</pre>

7) Даны числа x, y, x_1, y_1, x_2, y_2 . Проверить истинность высказывания: «Точка с координатами (x, y) лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты (x_1, y_1) , правая нижняя — (x_2, y_2) , а стороны параллельны координатным осям».

PascalABC.NET	C++
<pre>var x, y, x1, y1, x2, y2: Real; z: Boolean; begin WriteLn('(x,y)'); ReadLn(x, y); WriteLn('(x1,y1)'); ReadLn(x1, y1); WriteLn('(x2,y2)');</pre>	<pre>#include <iostream> using namespace std; int main(){ float x, y, x1, y1, x2, y2; cout << "(x, y): "; cin >> x >> y; cout << "(x1, y1): "; cin >> x1 >> y1; cout << "(x2, y2): "; cin >> x2 >> y2; bool w=((y < y1) && (y > y2) &&</pre>

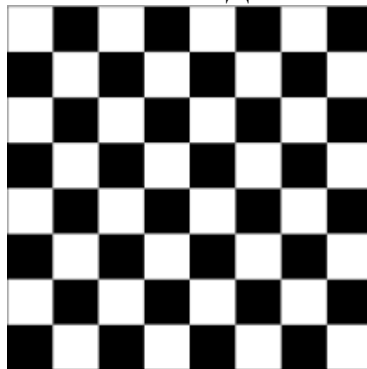
<pre> ReadLn(x2, y2); z:=(y<y1) And (y>y2) And (x > x1) And (x < x2); WriteLn(z); end.</pre>	<pre> (x > x1) && (x < x2)); cout <<w ; system("pause"); return 0; }</pre>
<p style="text-align: center;">Python</p> <pre> x= float(input("->")) y = float(input("->")) x1= float(input("->")) y1 = float(input("->")) x2= float(input("->")) x2 = float(input("->")) z = ((x1 < x) and (x<x2) and (y2<y) and (y<y1)) print(z)</pre>	<p style="text-align: center;">C#</p> <pre> static void Main(string[] args){ bool z; double x, x1, x2, y, y1, y2; x=Convert.ToDouble(Console.ReadLine()); x1=Convert.ToDouble(Console.ReadLine()); x2=Convert.ToDouble(Console.ReadLine()); y=Convert.ToDouble(Console.ReadLine()); y1=Convert.ToDouble(Console.ReadLine()); y2=Convert.ToDouble(Console.ReadLine()); z=((x1<x) && (x<x2) && (y2<y) && (y<y1)); Console.Write(z); Console.ReadKey(); }</pre>
<p style="text-align: center;">Java</p> <pre> public static void main(String[] args) { boolean L; int x,x1,x2,y,y1,y2; Scanner sc=new Scanner(System.in); x=sc.nextInt(); x1=sc.nextInt(); x2=sc.nextInt(); y=sc.nextInt(); y1=sc.nextInt(); y2=sc.nextInt(); L=(x1<x)&&(x<x2)&&(y1<y)&& (y<y2); System.out.println(L); }</pre>	<p style="text-align: center;">Go</p> <pre> package main import "fmt" func main() { fmt.Println("x,y,x1,x2,y1,y2") var x1,x2,y1,y2,x,y int var L bool fmt.Scanf("%d %d %d %d %d %d",&x1,&x2,&y1,&y2,&x,&y) L=(x1<x)&&(x<x2)&&(y1<y)&& (y<y2); fmt.Println("L=",L) }</pre>

8) Даны целые числа a , b , c , являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами a , b , c является равнобедренным».

<p style="text-align: center;">PascalABC.NET</p> <pre> Var x,y,x1,y1,x2,y2:Real; z: Boolean; begin WriteLn('(x,y)'); ReadLn(x, y); WriteLn('(x1,y1)'); ReadLn(x1, y1); WriteLn('(x2,y2)'); ReadLn(x2, y2); z :=(y<y1) And (y > y2) And (x>x1) And (x<x2); WriteLn(z); end.</pre>	<p style="text-align: center;">C++</p> <pre> #include <iostream> using namespace std; int main() { int a, b, c; cin >> a >> b >> c; bool s=((a==b) (b==c) (c==b)); cout << s; system("pause"); return 0; }</pre>
---	--

Python	C#
<pre>a= float(input("->")) b = float(input("->")) c= float(input("->")) z = (a==b) or (a==c) or (b==c) print(z)</pre>	<pre>static void Main(string[] args){ bool y; int a, b, c; a=Convert.ToInt32(Console.ReadLine()); b=Convert.ToInt32(Console.ReadLine()); c=Convert.ToInt32(Console.ReadLine()); y=((a==b) (b==c)&&(c==a)); Console.Write(y); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ boolean L; int a,b,c; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); c=sc.nextInt(); L=(a==b) (b==c) (c==a); System.out.println(L); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a,b,c->") var a,b,c int var L bool fmt.Scanf("%d %d %d",&a,&b,&c) L=(a==b) (b==c) (c==a); fmt.Println("L=",L) }</pre>

Boolean34. Даны координаты поля шахматной доски x, y (целые числа, лежащие в диапазоне 1–8). Учитывая, что левое нижнее поле доски (1, 1) является черным, проверить истинность высказывания: «Данное поле является белым».



PascalABC.NET	C++
<pre>Var var x, y: Integer; z: Boolean; begin ReadLn(x, y); z:=not ((y + x) Mod 2 = 0); WriteLn(z); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int x, y; cin >> x >> y; cout << !((x + y) % 2 == 0); system("pause"); return 0; }</pre>
Python	C#
<pre>x= int(input("->")) y = int(input("->")) z=(x+y)%2==1 print(z)</pre>	<pre>static void Main(string[] args){ bool k; int x, y; x= Convert.ToInt32(Console.ReadLine());</pre>

	<pre> y= Convert.ToInt32(Console.ReadLine()); k = (x+y) % 2 == 1; Console.Write(k); Console.ReadKey() } </pre>
Java	Go
<pre> public static void main(String[] args) { boolean L; int x,y; Scanner sc=new Scanner(System.in); x=sc.nextInt(); y=sc.nextInt(); L=(x+y) % 2 == 1; System.out.println(L); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("x,y") var x,y int var L bool fmt.Scanf("%d %d",&x,&y) L=(x+y) % 2 == 1; fmt.Println("L=",L) } </pre>

9) Даны координаты двух различных полей шахматной доски x_1, y_1, x_2, y_2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Ладья за один ход может перейти с одного поля на другое».

PascalABC.NET	C++
<pre> var x1, y1, x2, y2: Integer; z: Boolean; begin ReadLn(x1, y1); ReadLn(x2, y2); z:=(x1 = x2) Or (y1 = y2); WriteLn(z); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int x1, x2, y1, y2; cin >> x1 >> y1 >> x2 >> y2; cout<<((x1==x2) (y1==y2)); system("pause"); return 0; } </pre>
Python	C#
<pre> x1= int(input("->")) y1 = int(input("->")) x2= int(input("->")) y2 = int(input("->")) z = (x1 == x2) or (y1 == y2) print(z) </pre>	<pre> static void Main(string[] args){ int x1, x2, y1, y2; bool z; x1=Convert.ToInt32(Console.ReadLine()); y1=Convert.ToInt32(Console.ReadLine()); x2=Convert.ToInt32(Console.ReadLine()); y2=Convert.ToInt32(Console.ReadLine()); z = (x1 == x2) (y1 == y2); Console.WriteLine(z); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args){ boolean L; int x1, x2, y1, y2; Scanner sc=new Scanner(System.in); x1=sc.nextInt(); y1=sc.nextInt(); x2=sc.nextInt(); y2=sc.nextInt(); L=(x1 == x2) (y1 == y2); System.out.println(L); } </pre>	<pre> package main import "fmt" func main(){ fmt.Println("x1,x2,y1,y2") var x1,x2,y1,y2 int var L bool fmt.Scanf("%d %d %d %d",&x1, &x2,&y1,&y2) L=(x1 == x2) (y1 == y2); fmt.Println("L=",L) } </pre>

Задачи для самостоятельной работы

Во всех заданиях данной группы требуется вывести логическое значение True, если приведенное высказывание для предложенных исходных данных является истинным, и значение False в противном случае. Все числа, для которых указано количество цифр (двузначное число, трехзначное число и т. д.), считаются целыми положительными.

- 1) Дано целое число A . Проверить истинность высказывания: «Число A является нечетным».
- 2) Дано целое число A . Проверить истинность высказывания: «Число A является четным».
- 3) Даны два целых числа: A, B . Проверить истинность высказывания: «Справедливы неравенства $A > 2$ и $B < 3$ ».
- 4) Даны три целых числа: A, B, C . Проверить истинность высказывания: «Справедливо двойное неравенство $A < B < C$ ».
- 5) Даны три целых числа: A, B, C . Проверить истинность высказывания: «Число B находится между числами A и C ».
- 6) Даны два целых числа: A, B . Проверить истинность высказывания: «Хотя бы одно из чисел A и B нечетное».
- 7) Даны два целых числа: A, B . Проверить истинность высказывания: «Числа A и B имеют одинаковую четность».
- 8) Даны три целых числа: A, B, C . Проверить истинность высказывания: «Каждое из чисел A, B, C положительное».
- 9) Даны три целых числа: A, B, C . Проверить истинность высказывания: «Хотя бы одно из чисел A, B, C положительное».
- 10) Даны три целых числа: A, B, C . Проверить истинность высказывания: «Ровно одно из чисел A, B, C положительное».
- 11) Даны три целых числа: A, B, C . Проверить истинность высказывания: «Ровно два из чисел A, B, C являются положительными».
- 12) Дано целое положительное число. Проверить истинность высказывания: «Данное число является четным двузначным».
- 13) Дано целое положительное число. Проверить истинность высказывания: «Данное число является нечетным трехзначным».
- 14) Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара взаимно противоположных».
- 15) Дано трехзначное число. Проверить истинность высказывания: «Все цифры данного числа различны».
- 16) Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую последовательность».
- 17) Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую или убывающую последовательность».
- 18) Дано четырехзначное число. Проверить истинность высказывания: «Данное число читается одинаково слева направо и справа налево».
- 19) Даны числа A, B, C (число A не равно 0). Рассмотрев дискриминант $D = B^2 - 4 \cdot A \cdot C$, проверить истинность высказывания: «Квадратное уравнение $A \cdot x^2 + B \cdot x + C = 0$ имеет вещественные корни».
- 20) Даны числа x, y . Проверить истинность высказывания: «Точка с координатами (x, y) лежит в четвертой координатной четверти».

- 21) Даны числа x, y . Проверить истинность высказывания: «Точка с координатами (x, y) лежит во второй или третьей координатной четверти».
- 22) Даны числа x, y . Проверить истинность высказывания: «Точка с координатами (x, y) лежит в первой или третьей координатной четверти».
- 23) Даны целые числа a, b, c , являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами a, b, c является равносторонним».
- 24) Даны целые числа a, b, c , являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами a, b, c является прямоугольным».
- 25) Даны целые числа a, b, c . Проверить истинность высказывания: «Существует треугольник со сторонами a, b, c ».
- 26) Даны координаты двух различных полей шахматной доски x_1, y_1, x_2, y_2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Данные поля имеют одинаковый цвет».
- 27) Даны координаты двух различных полей шахматной доски x_1, y_1, x_2, y_2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Король за один ход может перейти с одного поля на другое».
- 28) Даны координаты двух различных полей шахматной доски x_1, y_1, x_2, y_2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Слон за один ход может перейти с одного поля на другое».
- 29) Даны координаты двух различных полей шахматной доски x_1, y_1, x_2, y_2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Ферзь за один ход может перейти с одного поля на другое».
- 30) Даны координаты двух различных полей шахматной доски x_1, y_1, x_2, y_2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Конь за один ход может перейти с одного поля на другое».
- 31) Вычислить значение логического выражения при следующих значениях логических величин A, B и C : $A = \text{Истина}$, $B = \text{Ложь}$, $C = \text{Ложь}$:
 а) A или B ; б) A и B ; в) B или C .
- 32) Вычислить значение логического выражения при заданных значениях логических величин A, B и C :
 а) не A и B ; б) A или не B ; в) A и B или C .
- 33) Даны вещественные числа x и y . Вычислить значение логического выражения
 а) $x^2 + y^2 \leq 4$;
 б) $x \geq 0$ или $y^2 \neq 4$;
 в) $x \cdot y \neq 0$ и $y > x$;
 г) не($x \cdot y < 0$) и $y > x$;
- 34) Даны два целых числа a и b . Если a делится на b или b делится на a , то вывести 1, иначе — любое другое число.
- 35) Три точки на плоскости заданы своими координатами. Выяснить, образуют ли эти точки треугольник.
- 36) Проверить, являются ли все цифры трехзначного числа различными.
- 37) Три точки на плоскости заданы своими координатами. Выяснить, образуют ли эти точки правильный треугольник.

- 38) Проверить, могут ли все цифры трехзначного числа образовать арифметическую прогрессию.
- 39) Четыре точки на плоскости заданы своими координатами. Выяснить, образуют ли эти точки квадрат.
- 40) Проверить, могут ли все цифры трехзначного числа образовать геометрическую прогрессию.
- 41) Даны три числа. Определить, являются ли эти числа сторонами прямоугольного треугольника.
- 42) Проверить, поместится ли на диске компьютера музыкальная композиция, которая длится m минут и n секунд, если свободное дисковое пространство b мегабайт, а для записи одной секунды звука необходимо 16 килобайт.
- 43) Координаты двух полей шахматной доски заданы в виде двух пар чисел x_1, y_1 и x_2, y_2 . На первом поле стоит ферзь, на втором - конь. Другу.
- 44) Две точки на плоскости $A(x_1; y_1)$ и $B(x_2; y_2)$ заданы своими координатами. Выяснить, которая из этих точек ближе к центру координатной плоскости.
- 45) Проверить, могут ли все цифры четырехзначного числа образовать стороны прямоугольника.
- 46) Две точки на плоскости $A(x_1; y_1)$ и $B(x_2; y_2)$ задано своими координатами. Проверить, лежат ли эти точки в одной координатной четверти!
- 47) Проверить, какому интервалу принадлежит данное действительное число d : $(-16, 2)$, $(4, 10)$ или $(14, 36)$. Если число d не принадлежит ни одному из интервалов, то выдать соответствующее сообщение.
- 48) Известен рост трех человек. Определить, одинаков ли их рост?
- 49) Даны вещественные положительные числа a, b, c, d . Выяснить, можно ли прямоугольник со сторонами a, b уместить внутри прямоугольника со сторонами c, d так, чтобы каждая из сторон одного прямоугольника была параллельна или перпендикулярна каждой стороне второго прямоугольника.
- 50) Вася пытается высунуть голову в форточку размерами a и b см. Приняв условно, что его голова — круглая диаметром d см, определить, сможет ли Вася сделать это. Для прохождения головы в форточку необходим зазор в 1 см с каждой стороны.

- 51) Даны вещественные положительные числа a, b, c, x, y . Выяснить, пройдет ли кирпич с ребрами a, b, c в прямоугольное отверстие со сторонами x и y . Просовывать кирпич в отверстие разрешается только так, чтобы каждое из его ребер было параллельно или перпендикулярно каждой из сторон отверстия.
- 52) Пассажир должен был сдать в камеру хранения пустой чемодан в форме параллелепипеда размерами a_1, a_2 и a_3 см и коробку размерами b_1, b_2 и b_3 см. Оплачивать нужно каждый размещаемый предмет. Определить, сможет ли пассажир сэкономить на оплате, поместив коробку в чемодан так, что стороны чемодана и коробки будут параллельны либо перпендикулярны друг другу.
- 53) Имеются стол прямоугольной формы с размерами $a \times b$ (a и b — целые числа, $a > b$) и кости домино с размерами $c \times d \times e$ (c, d и e — целые числа, $c > d > e$). Найти вариант размещения на столе наибольшего количества костей. Все размещаемые кости должны лежать на одной и той же грани в один ярус без свешивания со стола. Все ребра костей домино должны быть параллельны или перпендикулярны каждой стороне стола.

4. Условный оператор: группа If

1) Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае не изменять его. Вывести полученное число.

PascalABC.NET	C++
<pre> Var A: Integer; begin ReadLn(A); if A > 0 Then A := A + 1; WriteLn(A); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a; cin>>a; if (a>0) a++; cout<<a<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("->")) if (a>0): a=a+1 print(a) </pre>	<pre> static void Main(string[] args){ int a; a = Convert.ToInt32(Console.ReadLine()); if (a>0) a = a + 1; Console.WriteLine (a); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { int A; Scanner sc=new Scanner(System.in); A=sc.nextInt(); if(A>0) A=A+1; System.out.println(A); } </pre>	<pre> package main import "fmt" func main() { var a int fmt.Scanf("%d", &a) if a>0{ a+=1 } fmt.Printf("%d", a) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextInt() if (a>0) a++ println("a=\$a") } </pre>	
VB.NET	
<pre> Sub Main() Dim a As Integer a = Convert.ToInt32(Console.ReadLine()) If a > 0 Then a = a + 1 Console.WriteLine(a) Console.ReadKey() End Sub </pre>	

2) Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае вычесть из него 2. Вывести полученное число.

PascalABC.NET	C++
<pre> Var A: Integer; begin ReadLn(A); if A > 0 Then WriteLn(A + 1) Else WriteLn(A - 2); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a; cin>>a; if (a>0) a++; cout<<a<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("->")) if (a>0): a=a+1 else: a=a-2 print(a) </pre>	<pre> static void Main(string[] args){ int a; a = Convert.ToInt32(Console.ReadLine()); if (a > 0) a = a + 1; else a = a - 2; Console.WriteLine (a); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { int A; Scanner sc=new Scanner(System.in); A=sc.nextInt(); if(A>0) A=A+1; else A=A-2; System.out.println(A); } </pre>	<pre> package main import "fmt" func main() { var a int fmt.Scanf("%d", &a) if a>0{ a+=1 }else{ a=a-2 } fmt.Println("a=", a) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextInt() if (a>0) a++ else a-=2 println("a=\$a") } </pre>	
VB.NET	
<pre> Sub Main() Dim a As Integer a = Convert.ToInt32(Console.ReadLine()) </pre>	

```

    If a > 0 Then
        a = a + 1
    Else
        a = a - 2
    End If
    Console.WriteLine(a)
    Console.ReadKey()
End Sub

```

3) Дано целое число. Если оно является положительным, то прибавить к нему 1; если отрицательным, то вычесть из него 2; если нулевым, то заменить его на 10. Вывести полученное число.

PascalABC.NET	C++
<pre> Var A: Integer; begin ReadLn(A); if A > 0 Then WriteLn(A + 1) Else if A < 0 Then WriteLn(A - 2) Else WriteLn(10); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a; cin>>a; if (a>0) a++; else if (a<0) a-=2; else a=10; cout<<a<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("->")) if (a>0): a=a+1 elif (a<0): a=a-2 else: a=10 print(a) </pre>	<pre> static void Main(string[] args){ int a=Convert.ToInt32(Console.ReadLine()); if (a > 0) a++; else if (a < 0) a = a - 2; else a = 10; Console.WriteLine (a); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args){ int A; Scanner sc=new Scanner(System.in); A=sc.nextInt(); if(A>0){ A=A+1;} else if (A<0) A=A-2; else A=10; System.out.println(A); } </pre>	<pre> package main import "fmt" func main() { var a int fmt.Scanf("%d", &a) if a>0{ a+=1 }else if a<0{ a=a-2 }else{ a=10 } fmt.Println("a=", a) } </pre>

Kotlin
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var a=reader.nextInt() if (a>0) a++ else if (a<0) a-=2 else a=10 println("a=\$a") }</pre>
VB.NET
<pre>Sub Main() Dim a As Integer a = Convert.ToInt32(Console.ReadLine()) If a > 0 Then a = a + 1 ElseIf a < 0 Then a = a - 2 Else a = 10 End If Console.WriteLine(a) Console.ReadKey() End Sub</pre>

4) Даны три целых числа. Найти количество положительных чисел в исходном наборе.

PascalABC.NET	C++
<pre>Var A, B, C, k: Integer; begin k := 0; WriteLn('A,B, C: '); ReadLn(A, B, C); if A > 0 Then k := k + 1; if B > 0 Then k := k + 1; if C > 0 Then k := k + 1; WriteLn('k:', k); end.</pre>	<pre>#include <iostream> using namespace std; int main() { int A, B, C, k = 0; cout << "A, B, C: "; cin >> A >> B >> C; if(A > 0) k++; if(B > 0) k++; if(C > 0) k++; cout << "k: " << k << endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input("->")) b= int(input("->")) c= int(input("->")) k=0; if (a>0):</pre>	<pre>static void Main(string[] args){ int a, b, c, k=0; a=Convert.ToInt32(Console.ReadLine()); b=Convert.ToInt32(Console.ReadLine()); c=Convert.ToInt32(Console.ReadLine());</pre>

<pre> k+=1 if (b>0): k+=1 if (c>0): k+=1 print(k) </pre>	<pre> if (a > 0) k++; if (b > 0) k++; if (c>0) k++; Console.WriteLine (k); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { int a,b,c,k=0; Scanner sc=new Scanner (System.in); a=sc.nextInt(); b=sc.nextInt(); c=sc.nextInt(); if(a>0) k++; if(b>0) k++; if(c>0) k++; System.out.println(k); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("a,b,c->") var a,b,c int var k int fmt.Scanf("%d %d %d", &a,&b,&c) if(a>0){ k+=1 } if(b>0){ k++ } if(c>0){ k++ } fmt.Println("K=",k) } </pre>

5) Даны два числа. Вывести большее из них.

PascalABC.NET	C++
<pre> Var A, B: Integer; begin WriteLn('A, B: '); ReadLn(A, B); if A > B Then WriteLn(A) Else WriteLn(B); end. </pre>	<pre> Вариант 1. #include <iostream> using namespace std; int main() { int a,b; cin>>a>>b; if(a > b) cout<<a<<endl; else cout<<b<<endl; system("pause"); return 0; } //Вариант 2. #include <iostream> using namespace std; int main(){ int a,b,mx; cin>>a>>b; mx=a>b?a:b; cout<<mx<<endl; system("pause"); return 0; } </pre>

Python	C#
<pre>a= int(input("->")) b= int(input("->")) if (a>b): print(a) else: print(b)</pre>	<pre>static void Main(string[] args){ int a, b, mx; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); if (a > b) mx = a; else mx = b; Console.WriteLine (mx); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { int A,B; Scanner sc=new Scanner(System.in); A=sc.nextInt(); B=sc.nextInt(); if(A>B) System.out.println(A); else System.out.println(B); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("a,b->") var a,b,mx int fmt.Scanf("%d %d", &a,&b) if a > b { mx = a; } if a<b { mx = b; } fmt.Println("Max=",mx) }</pre>

6) Даны две переменные вещественного типа: A , B . Перераспределить значения данных переменных так, чтобы в A оказалось меньшее из значений, а в B — большее. Вывести новые значения переменных A и B .

PascalABC.NET	C++
<pre>Var A, B: Real; begin WriteLn('A, B: '); ReadLn(A, B); if A < B Then begin (A,B):=(B,A) end; WriteLn('A=', A, ' B=', B); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int a,b,h; cin>>a>>b; if (a<b){ h=a; a=b; b=h; } cout<<a<<" "<<b<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input("->")) b= int(input("->")) if (a>b): (a,b)=(b,a) print(a, ' ',b)</pre>	<pre>static void Main(string[] args){ int a, b, h; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); if(a>b){ c=b; b=a; a=c; } Console.WriteLine(a+" "+b); }</pre>

Java	Go
<pre>public static void main(String[] args){ int a,b,c; Scanner sc=new Scanner (System.in); a=sc.nextInt(); b=sc.nextInt(); if(a>b){ c=b; b=a; a=c; } System.out.println(a+" "+b); }</pre>	<pre>package main import "fmt" func main() { var a,b,c int fmt.Scanf("%d %d",&a,&b) if(a>b){ c=b; b=a; a=c;} fmt.Println("a=",a,"\nb=",b) }</pre>

7) Даны три числа. Найти наименьшее из них.

PascalABC.NET	C++
<pre>Var A, B, C, mn: Real; begin WriteLn('A,B, C: '); ReadLn(A, B, C); Mn := a; if mn < B Then Mn := B; if mn > C then Mn := c; WriteLn('Min = ', Mn) end.</pre>	<pre>#include <iostream> using namespace std; int main() { int a,b,c,mn; cin>>a>>b>>c; mn=a; if (mn<b) mn=b; if (mn<c) mn=c; cout<<mn<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input("->")) b= int(input("->")) c= int(input("->")) mn=a if (mn>b): mn=b if (mn>c): mn=c; print(mn)</pre>	<pre>static void Main(string[] args) { int a, b, c, mn; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); c = Convert.ToInt32(Console.ReadLine()); mn=a; if (mn<b) mn=b; if (mn<c) mn=c; Console.WriteLine(mn); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { int a,b,c,d; Scanner sc=new Scanner (System.in); a=sc.nextInt(); b=sc.nextInt(); c=sc.nextInt();</pre>	<pre>package main import "fmt" func main() { var a,b,c,mn int fmt.Scanf("%d %d %d", &a,&b,&c) mn=a;</pre>

<pre>d=a; if(d>b) c=b; if (d>c) d=c; System.out.println(d); }</pre>	<pre>if mn>b{ mn=b; } if mn>c { mn=c; } fmt.Println("Min=",mn) }</pre>
---	--

8) Даны три переменные вещественного типа: A, B, C . Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных A, B, C .

PascalABC.NET	C++
<pre>Var A, B, C: Real; begin WriteLn('A,B,C:'); ReadLn(A, B, C); if (A<B) And (B<C) Then begin A := A * 2; B := B * 2; C := C * 2; end Else begin A := -A; B := -B; C := -C; end; WriteLn('A=', A, ' B=', B, ' C = ', C); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int a,b,c; cin>>a>>b>>c; if ((a < b)&&(b < c)) { a*=2; b*=2; c*=2; } else { a*=-1; b*=-1; c*=-1; } cout<<a<<" "<<b<<" "<<c<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input("->")) b= int(input("->")) c= int(input("->")) if (a<b<c): a=a*2 b=b*2 c=c*2 else: a=-a b=-b c=-c print(a, ' ',b, ' ',c)</pre>	<pre>static void Main(string[] args){ int a, b, c; a=Convert.ToInt32(Console.ReadLine()); b=Convert.ToInt32(Console.ReadLine()); c=Convert.ToInt32(Console.ReadLine()); if ((a < b) && (b < c)) { a = a * 2; b = b * 2; c = c * 2; } else{ a=-a; b=-b; c=-c; } Console.WriteLine(a); Console.ReadKey(); }</pre>

Java	Go
<pre>public static void main(String[] args) { int a,b,c; Scanner sc=new Scanner (System.in); a=sc.nextInt(); b=sc.nextInt(); c=sc.nextInt(); if((a<b)&&(b<c)){ a=a*2; b=b*2; c=c*2; } else{ a=-a; b=-b; c=-c; } System.out.println(a+" "+b+" "+c); }</pre>	<pre>package main import "fmt" func main() { var a,b,c int fmt.Scanf("%d %d %d", &a,&b,&c) if((a<b)&&(b<c)){ a=a*2; b=b*2; c=c*2; }else{ a=-a; b=-b; c=-c; } fmt.Println("a=",a,"b=",b, "c=",c) }</pre>

9) Даны три целых числа, одно из которых отлично от двух других, равных между собой. Определить порядковый номер числа, отличного от остальных.

PascalABC.NET	C++
<pre>var A, B, C: Real; begin Readln(A, B, C); if B = C then Writeln('1'); if A = C then Writeln('2'); if A = B then Writeln('3'); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ SetConsoleCP(1251); SetConsoleOutputCP(1251); int a, b, c, n; cin >> a>>b>>c; if (b == c) n = 1; else if (a == c) n = 2; else if (a == b) n = 3; cout << n << endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= int(input("->")) b= int(input("->")) c= int(input("->")) k=0 if (b==c): k=1; if (a==c): k=2 if (a==b): k=3 print(k)</pre>	<pre>static void Main(string[] args){ int a, b, c; a=Convert.ToInt32(Console.ReadLine()); b=Convert.ToInt32(Console.ReadLine()); c=Convert.ToInt32(Console.ReadLine()); if ((a==b) && (b!=c)) Console.WriteLine(3); elseif ((a == c) && (c != b)) Console.WriteLine(2); else if ((c == b) && (b != a)) Console.WriteLine(1); Console.ReadKey(); }</pre>

Java	Go
<pre>public static void main(String[] args) { int a,b,c,h = 0; Scanner sc=new Scanner (System.in); a=sc.nextInt(); b=sc.nextInt(); c=sc.nextInt(); if((a==b)&&(b!=c)){ h=3; } if((a==c)&&(c!=b)){ h=2; } if((b==c)&&(c!=a)){ h=1; } System.out.println(h); }</pre>	<pre>package main import ("fmt") func main() { var a,b,c int fmt.Scanf("%d %d %d", &a,&b,&c) if((a==b)&&(b!=c)){ fmt.Println("N=",3) } if((a==c)&&(c!=b)){ fmt.Println("N=",2) } if((b==c)&&(c!=a)){ fmt.Println("N=",1) } }</pre>

10) На числовой оси расположены три точки: A , B , C . Определить, какая из двух последних точек (B или C) расположена ближе к A , и вывести эту точку и ее расстояние от точки A .

PascalABC.NET	C++
<pre>Var A, B, C: Real; begin WriteLn('A,B, C: '); ReadLn(A, B, C); if (B - A) < (C - A) Then WriteLn('B: ', (B - A)) Else WriteLn('C: ', (C - A)); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int c,b,a; cout << "a,b,c" ; cin >>a>>b>>c; if (abs(b-a)>abs(c-a)) cout<<"c-a:="<<abs(c-a); else cout<<"b-a:="<<abs(b-a); system("pause"); return 0; }</pre>
Python	C#
<pre>from math import sqrt a= int(input("->")) b= int(input("->")) c= int(input("->")) x=abs(b-a) y=abs(c-a) if (x<y): print('b:',x) elif (x>y): print('c:',y) else: print('AB=AC')</pre>	<pre>static void Main(string[] args){ int a, b, c, x, y; a=Convert.ToInt32(Console.ReadLine()); b=Convert.ToInt32(Console.ReadLine()); c=Convert.ToInt32(Console.ReadLine()); x= Math.abs(b-a); y = Math.abs(c - a); if (x<y) Console.WriteLine(b); else Console.WriteLine(a); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { double a,b,c,s;</pre>	<pre>package main import ("fmt" "math")</pre>

<pre> Scanner sc=new Scanner (System.in); a=sc.nextDouble(); b=sc.nextDouble(); c=sc.nextDouble(); if(b<c){ s=Math.abs(a-b); System.out.println(b+" "+s);} if(c<b){ s=Math.abs(a-c); System.out.println(c+" "+s); } </pre>	<pre> func main() { var a,b,c,s float64 fmt.Scanf("%v %v %v", &a,&b,&c) if(b<c){ s=math.Abs(a-b); fmt.Println("B=",s) } if(c<b){ s=math.Abs(a-c); fmt.Println("C=",s) } } </pre>
--	--

11) Даны целочисленные координаты трех вершин прямоугольника, стороны которого параллельны координатным осям. Найти координаты его четвертой вершины.

PascalABC.NET	C++
<pre> Var x1, y1, x2, y2, x3, y3, x4, y4: integer; begin ReadLn(x1, y1, x2, y2, x3, y3); if x1 = x2 Then x4 := x3; if x1 = x3 Then x4 := x2; if x2 = x3 Then x4 := x1; if y1 = y2 Then y4 := y3; if y1 = y3 Then y4 := y2; if y2 = y3 Then y4 := y1; WriteLn('(x4, y4: ', x4, ', ', y4); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int x1,y1,x2,y2,x3,y3,x4,y4; cin>> x1>>y1>>x2>>y2>>x3>>y3; if (x1==x2) x4=x3; if (x3==x2) x4=x1; if (x1==x3) x4=x2; if (y1==y2) y4=y3; if (y3==y2) y4=y1; if (y1==y3) y4=y2; cout<<x4<<" "<<y4<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> x1= int(input("->")) y1=int(input("->")) x2= int(input("->")) y2=int(input("->")) x3= int(input("->")) y3=int(input("->")) if (x1==x2): x4=x3; if (x3==x2): x4=x1 if (x1==x3): x4=x2 if (y1==y2): </pre>	<pre> static void Main(string[] args){ int x1,y1,x2,y2,x3,y3,x4,y4; x1=Convert.ToInt32(Console.ReadLine()); y1=Convert.ToInt32(Console.ReadLine()); x2=Convert.ToInt32(Console.ReadLine()); y2=Convert.ToInt32(Console.ReadLine()); x3=Convert.ToInt32(Console.ReadLine()); y3=Convert.ToInt32(Console.ReadLine()); if (x1==x2) x4=x3; if (x3==x2) x4=x1; if (x1==x3) x4=x2; if (y1==y2) y4=y3; if (y3==y2) y4=y1; </pre>

<pre> y4=y3; if (y3==y2): y4=y1 if (y1==y3): y4=y2 print(x4,' ',y4); </pre>	<pre> if (y1==y3) y4=y2; Console.WriteLine(x4+" "+y4); } </pre>
Java	Go
<pre> public static void main(String[] args) { int x1,y1,x2,y2,x3,y3,x4,y4; Scanner sc=new Scanner (System.in); x1=sc.nextInt(); y1=sc.nextInt(); x2=sc.nextInt(); y2=sc.nextInt(); x3=sc.nextInt(); y3=sc.nextInt(); if (x1==x2) x4=x3; if (x3==x2) x4=x1; if (x1==x3) x4=x2; if (y1==y2) y4=y3; if (y3==y2) y4=y1; if (y1==y3) y4=y2; System.out.println(x4+" "+y4); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("x1,y1,x2,y2,x3, y3") var x1,y1,x2,y2,x3,y3,x4,y4 int fmt.Scanf("%d %d %d %d %d %d", &x1,&y1,&x2,&y2,&x3,&y3) if (x1==x2) {x4=x3}; if (x3==x2) {x4=x1}; if (x1==x3) {x4=x2}; if (y1==y2) {y4=y3}; if (y3==y2) {y4=y1}; if (y1==y3) {y4=y2}; fmt.Println("x4=",x4,"\ny4=",y4) } </pre>

12) Для данного вещественного x найти значение следующей функции y , принимающей вещественные значения:

$$y = \begin{cases} -x, & \text{если } x < 0, \\ x^2, & \text{если } 0 < x < 2, \\ 4 & \text{если } x \geq 2. \end{cases}$$

PascalABC.NET	C++
<pre> Var x, y: Real; begin ReadLn(x); if x <= 0 Then y := -x Else if (x>0) and (x<2) Then y := Sqr(x) Else y := 4; WriteLn('f(x)=', y); end. </pre>	<pre> #include <cmath> #include <iostream> using namespace std; int main(){ float x,y; cin>> x; if (x<=0) y=-x; else if ((0<x) && (x<=2)) y=4; else y=sqrt(x); cout<<y<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> x= float(input("->")) if (x<=0): y=-x </pre>	<pre> static void Main(string[] args){ int y, x; x = Convert.ToInt32(Console.ReadLine()); </pre>

<pre> elif ((0<x) and (x<2)): y=x**2 else: y=4 print(y); </pre>	<pre> if(x<=0) f=-x; else if((0<x) && (x<2)) f=x*x; else f=4; Console.WriteLine(f); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { Double x,f = null; Scanner sc=new Scanner(System.in); x=sc.nextDouble(); if(x<=0) f=-x; else if((0<x) && (x<2)) f=x*x; else f=4; System.out.println(f); } </pre>	<pre> package main import ("fmt") func main() { var x,y int fmt.Scanf("%d",&x) if x<=0 { y=-x }else if x>0 && x<2 { y=x*x }else{ y=4 } fmt.Println("F=",y) } </pre>

13) Дан номер года (положительное целое число). Определить количество дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).

PascalABC.NET	C++
<pre> Var Year: Integer; begin ReadLn(Year); if Year Mod 4 = 0 Then begin if (Year Mod 100=0) And (Year Mod 400<>0) Then begin WriteLn('365') end Else WriteLn('366'); end Else WriteLn('365'); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int y; cin >>y; if (y % 4==0) { if ((y % 100==0)&& (y % 400)!=0)) cout << "365"<<<endl; else cout << "366"<<endl; } else cout << "365"<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> n= int(input("->")) if (n%4==0): if((n%100==0) and </pre>	<pre> static void Main(string[] args){ int n; n = Convert.ToInt32(Console.ReadLine()); </pre>

<pre> (n%400!=0)): print(365) else: print(366) else: print(365); </pre>	<pre> if (n % 4 == 0) if ((n % 100 == 0) && (n % 400 != 0)) Console.WriteLine(365); else Console.WriteLine(366); else Console.WriteLine(365); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { int y; Scanner sc=new Scanner(System.in); y=sc.nextInt(); if (y % 4==0){ if ((y%100==0)&& (y % 400)!=0)) System.out.println("365"); else System.out.println("366"); } else System.out.println("365"); } </pre>	<pre> package main import ("fmt") func main() { var y int fmt.Scanf("%d",&y) if (y % 4==0){ if((y%100==0)&&(y%400)!=0){ fmt.Println("365"); }else { fmt.Println("366"); }}else{ fmt.Println("365") } } </pre>

Задачи для самостоятельной работы

- 1) Даны три целых числа. Найти количество положительных и количество отрицательных чисел в исходном наборе.
- 2) Даны два числа. Вывести порядковый номер меньшего из них.
- 3) Даны два числа. Вывести вначале большее, а затем меньшее из них.
- 4) Даны две переменные целого типа: A и B . Если их значения не равны, то присвоить каждой переменной сумму этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных A и B .
- 5) Даны две переменные целого типа: A и B . Если их значения не равны, то присвоить каждой переменной большее из этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных A и B .
- 6) Даны три числа. Найти среднее из них (то есть число, расположенное между наименьшим и наибольшим).
- 7) Даны три числа. Вывести вначале наименьшее, а затем наибольшее из данных чисел.
- 8) Даны три числа. Найти сумму двух наибольших из них.
- 9) Даны три переменные вещественного типа: A , B , C . Если их значения упорядочены по возрастанию или убыванию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных A , B , C .
- 10) Даны четыре целых числа, одно из которых отлично от трех других, равных между собой. Определить порядковый номер числа, отличного от остальных.
- 11) Даны целочисленные координаты точки на плоскости. Если точка совпадает с началом координат, то вывести 0. Если точка не совпадает с началом координат, но лежит на оси OX или OY , то вывести соответственно 1 или 2. Если точка не лежит на координатных осях, то вывести 3.
- 12) Даны координаты точки, не лежащей на координатных осях OX и OY . Определить номер координатной четверти, в которой находится данная точка.
- 13) Для данного вещественного x найти значение следующей функции y , принимающей вещественные значения:

$$y = \begin{cases} 2 \cdot \sin(x), & \text{если } x > 0, \\ 6 - x, & \text{если } x \leq 0. \end{cases}$$

- 14) Для данного целого x найти значение следующей функции y , принимающей значения целого типа:

$$y = \begin{cases} 2 \cdot x, & \text{если } x < -2 \text{ или } x > 2, \\ -3 \cdot x, & \text{в противном случае.} \end{cases}$$

- 15) Для данного вещественного x найти значение следующей функции y , принимающей значения целого типа:

$$y = \begin{cases} 0, & \text{если } x < 0, \\ 1, & \text{если } x \text{ принадлежит } [0, 1), [2, 3), \dots, \\ -1, & \text{если } x \text{ принадлежит } [1, 2), [3, 4), \dots \end{cases}$$

- 16) Дано целое число. Вывести его строку-описание вида «отрицательное четное число», «нулевое число», «положительное нечетное число» и т. д.

- 17) Дано целое число, лежащее в диапазоне 1–999. Вывести его строку-описание вида «четное двузначное число», «нечетное трехзначное число» и т. д.

- 18) Три точки на плоскости заданы своими координатами. Выяснить, образуют ли эти точки треугольник.
- 19) Проверить, являются ли все цифры трехзначного числа различными.
- 20) Три точки на плоскости заданы своими координатами. Выяснить, образуют ли эти точки правильный треугольник.
- 21) Проверить, могут ли все цифры трехзначного числа образовать арифметическую прогрессию.
- 22) Четыре точки на плоскости заданы своими координатами. Выяснить, образуют ли эти точки квадрат.
- 23) Проверить, могут ли все цифры трехзначного числа образовать геометрическую прогрессию.
- 24) Даны три числа. Определить, являются ли эти числа сторонами прямоугольного треугольника.
- 25) Проверить, поместится ли на диске компьютера музыкальная композиция, которая длится m минут и n секунд, если свободное дисковое пространство 6 мегабайт, а для записи одной секунды звука необходимо 16 килобайт.
- 26) Координаты двух полей шахматной доски заданы в виде двух пар чисел x_1, y_1 и x_2, y_2 . На первом поле стоит ферзь, на втором - конь. Другу.
- 27) Две точки на плоскости $A(x_1; y_1)$ и $B(x_2; y_2)$ заданы своими координатами. Выяснить, которая из этих точек ближе к центру координатной плоскости.
- 28) Проверить, могут ли все цифры четырехзначного числа образовать стороны прямоугольника.
- 29) Известна цена монитора Samsung SyncMaster в январе 2000 г. и январе 2001 г. Ответьте на вопрос: "Произошло ли удешевление или нет? На сколько процентов изменилась цена изделия?"
- 30) Две точки на плоскости $A(x_1; y_1)$ и $B(x_2; y_2)$ задано своими координатами. Проверить, лежат ли эти точки в одной координатной четверти!
- 31) Проверить, какому интервалу принадлежит данное действительное число d : $(-16, 2)$, $(4, 10)$ или $(14, 36)$. Если число d не принадлежит не одному из интервалов, то выдать соответствующее сообщение.
- 32) Известен рост трех человек. Определить, одинаков ли их рост?
- 33) Даны вещественные положительные числа a, b, c, d . Выяснить, можно ли прямоугольник со сторонами a, b уместить внутри прямоугольника со сторонами c, d так, чтобы каждая из сторон одного прямоугольника была параллельна или перпендикулярна каждой стороне второго прямоугольника.
- 34) Вася пытается высунуть голову в форточку размерами a и b см. Приняв условно, что его голова — круглая диаметром d см, определить, сможет ли Вася сделать это. Для прохождения головы в форточку необходим зазор в 1 см с каждой стороны.
- 35) Даны вещественные положительные числа a, b, c, x, y . Выяснить, пройдет ли кирпич с ребрами a, b, c в прямоугольное отверстие со сторонами x и y .

Просовывать кирпич в отверстие разрешается только так, чтобы каждое из его ребер было параллельно или перпендикулярно каждой из сторон отверстия.

- 36) Пассажир должен был сдать в камеру хранения пустой чемодан в форме параллелепипеда размерами a_1, a_2 и a_3 см и коробку размерами b_1, b_2 и b_3 см. Оплачивать нужно каждый размещаемый предмет. Определить, сможет ли пассажир сэкономить на оплате, поместив коробку в чемодан так, что стороны чемодана и коробки будут параллельны либо перпендикулярны друг другу.
- 37) Имеются стол прямоугольной формы с размерами $a \times b$ (a и b — целые числа, $a > b$) и кости домино с размерами $c \times d \times e$ (c, d и e — целые числа, $c > d > e$). Найти вариант размещения на столе наибольшего количества костей. Все размещаемые кости должны лежать на одной и той же грани в один ярус без свешивания со стола. Все ребра костей домино должны быть параллельны или перпендикулярны каждой стороне стола.

5. Оператор выбора: группа Case

1) Дано целое число в диапазоне 1–7. Вывести строку — название дня недели, соответствующее данному числу (1 — «понедельник», 2 — «вторник» и т. д.).

PascalABC.NET	C++
<pre> Var x: Integer; begin ReadLn(x); case x Of 1: WriteLn('Monday'); 2: WriteLn('Tuesday'); 3: WriteLn('Wednesday'); 4: WriteLn('Thursday'); 5: WriteLn('Friday'); 6: WriteLn('Shanbe'); 7: WriteLn('Sunday') else Writeln('Error'); End; end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int x; cin >> x; switch (x){ case 1:cout << "Monday" << endl; break; case 2:cout << "Tuesday" << endl; break; case 3:cout<<"Wednesday " << endl; break; case 4:cout <<"Thursday"<<endl; break; case 5:cout << "Friday" << endl; break; case 6:cout << "Shanbe" << endl; break; case 7:cout <<"Sunday"<<endl; break; default:cout<<"Error"; } system("pause"); return 0; } </pre>
Python	C#
<pre> def switch_case(case): return { 1 : "Monday", 2 : "Tuesday", 3 : "Wednesday ", 4: "Thursday", 5 : "Friday", 6 : "Saturday", 7 : "Sunday" }.get(case, "Error") n= int(input("->")) print(switch_case(n)) </pre>	<pre> static void Main(string[] args){ int n; String S=""; n = Convert.ToInt32(Console.ReadLine()); switch (n){ case 1: S="Monday"; break; case 2: S="Tuesday"; break; case 3: S="Wednesday ";break; case 4: S="Thursday";break; case 5: S="Friday"; break; case 6: S="Shanbe"; break; case 7: S="Sunday";break; default: S="Error");break; } Console.WriteLine(S); } </pre>
Java	
<pre> public static void main(String[] args) { int n; Scanner sc=new Scanner(System.in); n=sc.nextInt(); switch(n){ case 1:System.out.println("Monday");break; </pre>	

<pre> case 2: System.out.println("Tuesday"); break; case 3: System.out.println("Wednesday "); break; case 4: System.out.println("Thursday"); break; case 5: System.out.println("Friday"); break; case 6: System.out.println("Shanbe"); break; case 7: System.out.println("Sunday"); break; default: System.out.println("Error!"); break; } </pre>	
Go	
<pre> package main import "fmt" func main() { fmt.Println("a=?") var a int fmt.Scanf("%d", &a) switch a { case 1: fmt.Println("Monday") case 2: fmt.Println("Tuesday") case 3: fmt.Println("Wednesday ") case 4: fmt.Println("Thursday") case 5: fmt.Println("Friday") case 6: fmt.Println("Shanbe") case 7: fmt.Println("Sunday") default: fmt.Println("Error") } } </pre>	
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var n=reader.nextInt() var S:String when(n) { 1 -> S = "Monday" 2 -> S = "Tuesday" 3 -> S = "Wednesday " 4 -> S = "Thursday" 5 -> S = "Friday"; 6 -> S = "Shanbe" 7 -> S = "Sunday" else-> S = "Error" } println(S) } </pre>	

VB.NET
<pre> Sub Main() Dim a As Integer Dim S As String a = Convert.ToInt32(Console.ReadLine()) Select Case a Case 1 : S = "Monday" Case 2 : S = "Tuesday" Case 3 : S = "Wednesday " Case 4 : S = "Thursday" Case 5 : S = "Friday" Case 6 : S = "Shanbe" Case 7 : S = "Sunday" Case Else : S = "Error" End Select Console.WriteLine(S) Console.ReadKey() End Sub </pre>

2) Дан номер месяца — целое число в диапазоне 1–12 (1 — январь, 2 — февраль и т. д.). Вывести название соответствующего времени года («зима», «весна», «лето», «осень»).

PascalABC.NET	C++
<pre> Var M: Integer; begin Write(' month: '); Readln(M); case M of 12, 1, 2: Writeln('Zima'); 3..5: Writeln('Vesna'); 6..8: Writeln('Leto'); 9..11: Writeln('Osen'); end; end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int month; cin>> month; switch (month) { case 12:case 1:case 2: cout<<"Zima\n" ; break; case 3:case 4:case 5: cout<<"Vesna\n" ; break; case 6:case 7:case 8: cout<<"Leto\n" ; break; case 9:case 10:case 11: cout<<"Osen\n" ; break; } system("pause"); return 0; } </pre>
Python	C#
<pre> def switch_case(case): return { 12:"Zima", 1:"Zima", 2:"Zima", 3:"Vesna", 4: "Vesna", </pre>	<pre> static void Main(string[] args){ int n; n= Convert.ToInt32(Console.ReadLine()); switch(n){ case 12: case 1: case 2:Console.WriteLine("Zima"); </pre>

<pre> 5: "Vesna", 6:"Leto", 7:"Leto", 8:"Leto", 9:"Osen", 10:"Osen", 11:"Osen", }.get(case, "an out of range number") n= int(input("->")) print(switch_case(n)) </pre>	<pre> break; case 3: case 4: case 5:Console.WriteLine ("Vesna"); break; case 6: case 7: case 8:Console.WriteLine("Leto"); break; case 9: case 10: case 11:Console.WriteLine ("Osen"); break; default:Console.WriteLine ("Error"); break; } Console.ReadKey (); } </pre>
Java	
<pre> public static void main(String[] args) { int k; Scanner sc=new Scanner(System.in); k=sc.nextInt(); switch(k){ case 12: case 1: case 2:System.out.println("Zima");break; case 3:case 4: case 5:System.out.println("Vesna");break; case 6:case 7: case 8:System.out.println("Leto");break; case 9:case 10: case 11:System.out.println("Osen");break; default:System.out.println("Error!");break; } } </pre>	
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var n=reader.nextInt() var S="" when(n) { 12,1,2 -> S = "Zima" 3, 4,5 -> S = "Vesna" in 6..8 -> S = "Leto " 9,10,11 -> S = "Osen" !in 1..12 -> S = "Error" } println(S) } } </pre>	

3) Арифметические действия над числами пронумерованы следующим образом: 1 — сложение, 2 — вычитание, 3 — умножение, 4 — деление. Дан номер действия

N (целое число в диапазоне 1–4) и вещественные числа A и B (B не равно 0).
 Выполнить над числами указанное действие и вывести результат.

PascalABC.NET	C++
<pre> var N, A, B: Integer; begin Write('N,A,B: '); Readln(N, A, B); case N of 1: Writeln(A + B); 2: Writeln(A - B); 3: Writeln(A * B); 4: Writeln(A / B); end; end. </pre>	<pre> #include <iostream> using namespace std; int main() { int n; float a,b; cin>>n>>a>>b; switch (n) { case 1: cout<<a+b<<endl; break; case 2: cout<<a-b<<endl; break; case 3: cout<<a*b<<endl; break; case 4: cout<<a/b<<endl; break; } system("pause"); return 0; } </pre>
C#	
<pre> static void Main(string[] args){ int n; double a, b; n = Convert.ToInt32(Console.ReadLine()); a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); switch(n) { case 1:Console.WriteLine(a+b);break; case 2: Console.WriteLine(a-b); break; case 3: Console.WriteLine(a*b); break; case 4: Console.WriteLine(a/b); break; default: Console.WriteLine ("Error"); break; } Console.ReadKey (); } </pre>	
Java	
<pre> public static void main(String[] args) { int k,a,b; Scanner sc=new Scanner(System.in); k=sc.nextInt(); a=sc.nextInt(); b=sc.nextInt(); } </pre>	

```

switch(k){
    case 1:System.out.println(a+b);break;
    case 2:System.out.println(a-b);break;
    case 3:System.out.println(a*b);break;
    case 4:System.out.println(a/b);break;
    default:System.out.println("Error");break;
}

```

5) Даны два целых числа: *D* (день) и *M* (месяц), определяющие правильную дату невисокосного года. Вывести значения *D* и *M* для даты, предшествующей указанной.

PascalABC.NET	C++
<pre> var D, M: Integer; begin Write('D,M: '); Readln(D, M); D := D - 1; if D = 0 then begin M := M - 1; if M = 0 then M := 12; case M of 1,3,5,7,8,10,12: D:= 31; 4, 6, 9, 11: D := 30; 2: D := 28; end; end; Writeln('D.M: ', D, '.', M); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int d, m; cout << "d,m \n"; cin >> d>>m; if (d == 1) { switch (m) { case 1: d = 31;break; case 3: d = 31;break; case 5: d = 31;break; case 7: d = 31;break; case 8: d = 31;break; case 10: d = 31;break; case 12: d = 31;break; case 4: d = 30;break; case 6: d = 30;break; case 9: d = 30;break; case 11: d = 30;break; case 2: d = 28;break; } } else d=d-1; cout<<d<<":"<<m<<endl; system("pause"); return 0; } </pre>
C#	
<pre> static void Main(string[] args){ int d, m; d = Convert.ToInt32(Console.ReadLine()); m = Convert.ToInt32(Console.ReadLine()); if (d == 1) switch (m){ case 1: Console.WriteLine(31 + " " + 12); break; case 2: Console.WriteLine(31 + " " + 1); break; case 3: Console.WriteLine(28 + " " + 2); break; case 4: Console.WriteLine(31 + " " + 3); break; case 5: Console.WriteLine(30 + " " + 4); break; case 6: Console.WriteLine(31 + " " + 5); break; case 7: Console.WriteLine(30 + " " + 6); break; case 8: Console.WriteLine(31 + " " + 7); break; </pre>	

```

        case 9: Console.WriteLine(31 + " " + 8); break;
        case 10: Console.WriteLine(30 + " " + 9); break;
        case 11: Console.WriteLine(31 + " " + 10); break;
        case 12: Console.WriteLine(30 + " " + 11); break;
    }
    else
        Console.WriteLine(d-1+ " " + m);
    Console.ReadKey();
}
}

```

Java

```

public static void main(String[] args) {
    int D,M;
    Scanner sc=new Scanner(System.in);
    D=sc.nextInt();
    M=sc.nextInt();
    if(D==1)
        switch(M){
            case 1:System.out.println(31+" "+12);break;
            case 2:System.out.println(31+" "+1);break;
            case 3:System.out.println(28+" "+2);break;
            case 4:System.out.println(31+" "+3);break;
            case 5:System.out.println(30+" "+4);break;
            case 6:System.out.println(31+" "+5);break;
            case 7:System.out.println(30+" "+6);break;
            case 8:System.out.println(31+" "+7);break;
            case 9:System.out.println(31+" "+8);break;
            case 10:System.out.println(30+" "+9);break;
            case 11:System.out.println(31+" "+10);break;
            case 12:System.out.println(30+" "+11);break;}
        else{
            D=D-1;
            System.out.println(D+" "+M);
        }
}

```

6) Робот может перемещаться в четырех направлениях («С» — север, «З» — запад, «Ю» — юг, «В» — восток) и принимать три цифровые команды: 0 — продолжать движение, 1 — поворот налево, -1 — поворот направо. Дан символ *C* — исходное направление робота и целое число *N* — посланная ему команда. Вывести направление робота после выполнения полученной команды.

PascalABC.NET	C++
<pre> Var C: Char; N: Integer; begin Write('C:'); Readln(C); Write('N:'); Readln(N); case C of 'N': </pre>	<pre> #include <iostream> using namespace std; int main() { int n; char c; cin>>c>>n; if (c=='C') switch (n) { case 1: c='З'; break; case 0:c='C'; break; case -1: c='В'; break; </pre>

<pre> case N of 1: C := 'W'; 0: C := 'N'; -1: C:='E'; end; 'E': case N of 1: C := 'N'; 0: C := 'E'; -1: C:='S'; end; 'S': case N of 1: C := 'E'; 0: C := 'S'; -1: C:='W'; end; 'W': case N of 1: C := 'S'; 0: C := 'W'; -1: C:='N'; end; end; Writeln(C); end. </pre>	<pre> } else if (c=='B') switch (n) { case 1: c='C'; break; case 0:c='B';break; case -1:c='0';break; } else if (c=='0') switch (n) { case 1:c='B'; break; case 0:c='0'; break; case -1: c='3';break; } else if (c=='3') switch (n) { case 1: c='0';break; case 0:c='3'; break; case -1: c='C'; break; } } cout<<c<<endl; system("pause"); return 0; } </pre>
C#	
<pre> static void Main(string[] args){ int n; char S; S = Convert.ToChar(Console.ReadLine()); n = Convert.ToInt32(Console.ReadLine()); if (S == 's') { switch (n) { case 0: Console.WriteLine("Sever"); break; case 1: Console.WriteLine("Zapad"); break; case -1: Console.WriteLine("Vostok"); break; } } else if (S == 'z'){ switch (n){ case 0: Console.WriteLine("Zapad"); break; case 1: Console.WriteLine("Yug"); break; case -1: Console.WriteLine("Sever"); break; } } else if (S == 'v'){ switch (n){ case 0: Console.WriteLine("Vostok"); break; case 1: Console.WriteLine("Sever"); break; } } } </pre>	

```

        case -1: Console.WriteLine("Yug"); break;
    }
}
else if (S == 'y')
{
    switch (n)
    {
        case 0: Console.WriteLine("Yug"); break;
        case 1: Console.WriteLine("Vostok"); break;
        case -1: Console.WriteLine("Zapad"); break;
    }
}
Console.ReadKey();
}

```

Java

```

public static void main(String[] args) {
    int n;
    String c, c1=null;
    Scanner sc=new Scanner(System.in);
    n=sc.nextInt();
    c= sc.next();
    if(c.equals("z"))
        switch(n){
            case 0: c1="z";break;
            case 1: c1="v";break;
            case -1: c1="y";break;
        }
    else if(c.equals("s"))
        switch(n){
            case 0: c1="s";break;
            case 1: c1="z";break;
            case -1: c1="y";break;
        }
    else if(c.equals("y"))
        switch(n){
            case 0: c1="y";break;
            case 1: c1="s";break;
            case -1: c1="v";break;
        }
    else if(c.equals("v"))
        switch(n){
            case 0: c1="v";break;
            case 1: c1="y";break;
            case -1: c1="z";break;
        }
    System.out.println(c1);
}

```

7) Элементы окружности пронумерованы следующим образом: 1 — радиус R , 2 — диаметр $D = 2 \cdot R$, 3 — длина $L = 2 \cdot \pi \cdot R$, 4 — площадь круга $S = \pi \cdot R^2$. Дан номер

одного из этих элементов и его значение. Вывести значения остальных элементов данной окружности (в том же порядке). В качестве значения π использовать 3.14.

PascalABC.NET	C++
<pre> Var h, R, D, L, S: real; n: integer; begin read(n); read(h); case n of 1: R := h; 2: R := h / 2; 3: R:=h/2*3.14; 4: R:=sqrt(h/3.14); end; D := 2 * R; L := 2 * 3.14; S := 3.14 * R * R; Writeln(R, ' ', D, ' ', L, ' ', S); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int n; double h,R=0,D,L,S; cin>>n>>h; switch(n){ case 1:R=h;break; case 2:R=h/2;break; case 3:R=h/2*3.14;break; case 4:R= sqrt(h/3.14);break; } D=2*R; L=2*3.14; S=3.14*R*R; cout<<R<<" "<<D<<" "; cout<<<<L<<" "<<S<<endl; system("pause"); return 0; } </pre>
C#	
<pre> static void Main(string[] args){ Double h, D, R, L, S; int n; h = Convert.ToDouble(Console.ReadKey()); n = Convert.ToInt32(Console.ReadLine()); switch (n) { case 1: R = h; break; case 2: R = h / 2; break; case 3: R = h / 2 * 3.14; break; case 4: R = Math.Sqrt(h / 3.14); break; } D = 2 * R; L = 2 * 3.14 * R; S = 3.14 * (R * R); Console.WriteLine("R={0},D={1},L={2},S={3}", R,D,L,S); Console.ReadKey(); } </pre>	
Java	
<pre> public static void main(String[] args) { int n; double h,R=0,D,L,S; Scanner sc=new Scanner(System.in); n=sc.nextInt(); h=sc.nextDouble(); switch(n){ case 1:R=h;break; case 2:R=h/2;break; </pre>	

```

        case 3:R=h/2*Math.PI;break;
        case 4:R=Math.sqrt(h/Math.PI);break;
    }
    D=2*R;
    L=2*Math.PI*R;
    S=Math.PI*R*R;
    System.out.println(R+" "+" "+D+" "+"L+" "+"S+" ");
}

```

7) Мастям игральных карт присвоены порядковые номера: 1 — пики, 2 — трефы, 3 — бубны, 4 — червы. Достоинству карт, старших десятки, присвоены номера: 11 — валет, 12 — дама, 13 — король, 14 — туз. Даны два целых числа: N — достоинство ($6 \leq N \leq 14$) и M — масть карты ($1 \leq M \leq 4$). Вывести название соответствующей карты вида «шестерка бубен», «дама червей», «туз треф» и т. п.

PascalABC.NET
<pre> var N, M: Integer; begin Write('Введите достоинство карты: '); Readln(N); Write('Введите масть карты: '); Readln(M); case N of 6: write('шестёрка '); 7: write('семёрка '); 8: write('восьмёрка '); 9: write('девятка '); 10: write('десятка '); 11: write('валет '); 12: write('дама '); 13: write('король '); 14: write('туз '); end; case M of 1: write('пик'); 2: write('треф'); 3: write('бубен'); 4: write('червей'); end; end. </pre>
C++
<pre> #include <iostream> using namespace std; int main(){ int n,m; cin>>n>>m; switch (n) { case 6: cout<<"шестёрка";break; case 7: cout<<"семёрка "; break; case 8: cout<<"восьмёрка "; break; case 9: cout<<"девятка ";break; case 10:cout<<"десятка "; break; </pre>

```

    case 11:cout<<"валет "; break;
    case 12: cout<<"дама ";break;
    case 13:cout<<"король ";break;
    case 14:cout<<"туз ";break;
    }
    switch (m) {
    case 1:cout<<"пик\n"; break;
    case 2:cout<<"треф\n"; break;
    case 3:cout<<"бубен\n";
        break;
    case 4:cout<<"червы\n";
        break;
    }
    system("pause");
    return 0;
}

```

8) Дано целое число в диапазоне 20–69, определяющее возраст (в годах). Вывести строку-описание указанного возраста, обеспечив правильное согласование числа со словом «год», например: 20 — «двадцать лет», 32 — «тридцать два года», 41 — «сорок один год».

```

PascalABC.NET
Var
    Y: Integer;

begin
    Write('Введите возраст: ');
    Readln(Y);
    case Y div 10 of
        2: write('двадцать ');
        3: write('тридцать ');
        4: write('сорок ');
        5: write('пятьдесят ');
        6: write('шестьдесят ');
    end;
    case Y mod 10 of
        1: write('один ');
        2: write('два ');
        3: write('три ');
        4: write('четыре ');
        5: write('пять ');
        6: write('шесть ');
        7: write('семь ');
        8: write('восемь ');
        9: write('девять ');
    end;

    case Y mod 10 of
        0, 5, 6, 7, 8, 9: write('лет. ');
        1: write('год. ');
        2, 3, 4: write('года. ');
    end;

end.

```

```

#include <iostream>
#include <windows.h>
using namespace std;
int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int y;
    cin >> y;
    switch (y / 10) {
        case 2:    cout << "двадцать "; break;
        case 3:    cout << "тридцать "; break;
        case 4:    cout << "сорок "; break;
        case 5:    cout << "пятьдесят "; break;
        case 6:    cout << "шестьдесят "; break;
    }
    switch (y % 10)
    {
        case 1:    cout << "один "; break;
        case 2:    cout << "два "; break;
        case 3:    cout << "три "; break;
        case 4:    cout << "четыре "; break;
        case 5:    cout << "пять "; break;
        case 6:    cout << "шесть "; break;
        case 7:    cout << "семь "; break;
        case 8:    cout << "восемь "; break;
        case 9:    cout << "девять "; break;
    }
    switch (y % 10)
    {
        case 0:
        case 5:
        case 6:
        case 7:
        case 8:
        case 9:
            cout << "лет.\n "; break;
        case 1: cout << "год.\n "; break;
        case 2:
        case 3:
        case 4:
            cout << "года.\n "; break;
    }
    system("pause");
    return 0;
}

```

Case20. Даны два целых числа: D (день) и M (месяц), определяющие правильную дату. Вывести знак Зодиака, соответствующий этой дате: «Водолей» (20.1–18.2), «Рыбы» (19.2–20.3), «Овен» (21.3–19.4), «Телец» (20.4–20.5), «Близнецы» (21.5–21.6), «Рак» (22.6–22.7), «Лев» (23.7–22.8), «Дева» (23.8–22.9), «Весы» (23.9–22.10), «Скорпион» (23.10–22.11), «Стрелец» (23.11–21.12), «Козерог» (22.12–19.1).

```

Var
  D, M: Integer;

begin
  Write('Введите День: ');
  Readln(D);
  Write('Введите Месяц: ');
  Readln(M);

  case M * 100 + D of
    0120..0218: Writeln('Водолей');
    0219..0320: Writeln('Рыбы');
    0321..0419: Writeln('Овен');
    0420..0520: Writeln('Телец');
    0521..0621: Writeln('Близнецы');
    0622..0722: Writeln('Рак');
    0723..0822: Writeln('Лев');
    0823..0922: Writeln('Дева');
    0923..1022: Writeln('Весы');
    1023..1122: Writeln('Скорпион');
    1123..1221: Writeln('Стрелец');
    1222..1231, 0101..0119: Writeln('Козерог');
  end;
end.

```

C++

```

#include <iostream>
#include <windows.h>
using namespace std;
int main() {
  SetConsoleCP(1251);
  SetConsoleOutputCP(1251);
  int d, m;
  cout << "день:";
  cin >> d;
  cout << "месяц:";
  cin >> m;
  switch (m) {
  case 1:
    if (d > 19) cout << "Водолей";
    else cout << "Козерог";
    break;
  case 2:
    if (d > 18) cout << "Рыбы";
    else cout << "Водолей";
    break;
  case 3:
    if (d > 20) cout << "Овен";
    else cout << "Рыбы";
    break;
  case 4:
    if (d > 19) cout << "Телец";
    else cout << "Овен";
    break;
  }
}

```

```
case 5:
    if (d > 20) cout << "Близнецы";
    else cout << "Телец";
    break;
case 6:
    if (d > 21) cout << "Рак";
    else cout << "Близнецы";
    break;
case 7:
    if (d > 22) cout << "Лев";
    else cout << "Рак";
    break;
case 8:
    if (d > 22) cout << "Дева";
    else cout << "Лев";
    break;
case 9:
    if (d > 22) cout << "Весы";
    else cout << "Дева";
    break;
case 10:
    if (d > 22) cout << "Скорпион";
    else cout << "Весы";
    break;
case 11:
    if (d > 22) cout << "Стрелец";
    else cout << "Скорпион";
    break;
case 12:
    if (d > 21) cout << "Козерог";
    else cout << "Стрелец";
    break;
}
cout << endl;
system("pause");
return 0;
}
```


Задачи для самостоятельной работы

- 1) Дано целое число K . Вывести строку-описание оценки, соответствующей числу K (1 — «плохо», 2 — «неудовлетворительно», 3 — «удовлетворительно», 4 — «хорошо», 5 — «отлично»). Если K не лежит в диапазоне 1–5, то вывести строку «ошибка».
- 2) Дан номер месяца — целое число в диапазоне 1–12 (1 — январь, 2 — февраль и т. д.). Определить количество дней в этом месяце для невисокосного года.
- 3) Единицы длины пронумерованы следующим образом: 1 — дециметр, 2 — километр, 3 — метр, 4 — миллиметр, 5 — сантиметр. Дан номер единицы длины (целое число в диапазоне 1–5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах.
- 4) Единицы массы пронумерованы следующим образом: 1 — килограмм, 2 — миллиграмм, 3 — грамм, 4 — тонна, 5 — центнер. Дан номер единицы массы (целое число в диапазоне 1–5) и масса тела в этих единицах (вещественное число). Найти массу тела в килограммах.
- 5) Даны два целых числа: D (день) и M (месяц), определяющие правильную дату невисокосного года. Вывести значения D и M для даты, следующей за указанной.
- 6) Локатор ориентирован на одну из сторон света («С» — север, «З» — запад, «Ю» — юг, «В» — восток) и может принимать три цифровые команды поворота: 1 — поворот налево, -1 — поворот направо, 2 — поворот на 180° . Дан символ C — исходная ориентация локатора и целые числа N_1 и N_2 — две посланные команды. Вывести ориентацию локатора после выполнения этих команд.
- 7) Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 — катет a , 2 — гипотенуза $c = a\sqrt{2}$, 3 — высота h , опущенная на гипотенузу ($h = c/2$), 4 — площадь $S = c \cdot h/2$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).
- 8) Элементы равностороннего треугольника пронумерованы следующим образом: 1 — сторона a , 2 — радиус R_1 вписанной окружности ($R_1 = \frac{a\sqrt{3}}{6}$), 3 — радиус R_2 описанной окружности ($R_2 = 2 \cdot R_1$), 4 — площадь $S = \frac{a^2\sqrt{3}}{4}$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).
- 9) Дано целое число в диапазоне 10–40, определяющее количество учебных заданий по некоторой теме. Вывести строку-описание указанного количества заданий, обеспечив правильное согласование числа со словами «учебное задание», например: 18 — «восемнадцать учебных заданий», 23 — «двадцать три учебных задания», 31 — «тридцать одно учебное задание».
- 10) Дано целое число в диапазоне 100–999. Вывести строку-описание данного числа, например: 256 — «двести пятьдесят шесть», 814 — «восемьсот четырнадцать».
- 11) В восточном календаре принят 60-летний цикл, состоящий из 12-летних подциклов, обозначаемых названиями цвета: зеленый, красный, желтый, белый и черный. В каждом подцикле годы носят названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. По номеру года определить его название, если 1984 год — начало цикла: «год зеленой крысы».

6. Цикл с параметром: группа For

1) Даны целые числа K и N ($N > 0$). Вывести N раз число K .

PascalABC.NET	C++
<pre> Var K, N, i: Integer; begin Write('N,K: '); Readln(N, K); for i := 1 to N do Writeln(K); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int k, n, i; cin>>k>>n; for (i = 0; i<n; ++i) cout<<k; system("pause"); return 0; } </pre>
Python	C#
<pre> n= int(input("->")) k= int(input("->")) for i in range(1,n): print(k) </pre>	<pre> static void Main(string[] args){ int n,k, i; n = Convert.ToInt32(Console.ReadLine()); k = Convert.ToInt32(Console.ReadLine()); for (i = 0; i <= n; i++) Console.WriteLine(k); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { int i,n,k; Scanner sc=new Scanner(System.in); n=sc.nextInt(); k=sc.nextInt(); for(i=1; i<=n; i++) System.out.print(k); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("n,k=?") var n,k,i int fmt.Scanf("%d %d",&n,&k) for i=1;i<=n; i++){ fmt.Print(k, " ") } } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var k = reader.nextInt() var n = reader.nextInt() for (i in 1..n) print("\$k ") } </pre>	
Вывод четных чисел	Обратный цикл
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner (System.`in`) var n = reader.nextInt() for (i in 2..n <i>step</i> 2) print("\$i ") } </pre>	<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var n = reader.nextInt() for (i in n <i>downTo</i> 1) print("\$i ") } </pre>

VB.NET
<pre> Sub Main() Dim n, k, i As Integer n = Convert.ToInt32(Console.ReadLine()) k = Convert.ToInt32(Console.ReadLine()) For i = 1 To n Console.Write(k) Next Console.ReadKey() End Sub </pre>
<pre> For number As Double = 2 To 0 Step -0.25 </pre>

2) Даны два целых числа A и B ($A < B$). Вывести в порядке возрастания все целые числа, расположенные между A и B (включая сами числа A и B), а также количество N этих чисел.

PascalABC.NET	C++
<pre> Var A, B, i: Integer; begin Write('A,B: '); Readln(A, B); for i:= 0 to B - A do Writeln(A + i); Writeln(B - A + 1); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a, b, i; cin>>a>>b; for (i = a; i<=b; ++i) cout<<i<<" "; cout<< b-a+1; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("->")) b= int(input("->")) k=0; for i in range(a,b): print(i) k+=1; print('k=',k) </pre>	<pre> static void Main(string[] args){ int a, b,i, k=0; a=Convert.ToInt32(Console.ReadLine ()); b=Convert.ToInt32 (Console.ReadLine ()); for (i = a; i <= b; i++){ k++; Console.Write(i + " "); } Console.WriteLine("\n k="+k); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args){ int i,a,b,k=0; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); for(i=a; i<=b; i++){ k++; System.out.print(i+" "); } System.out.println(k); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("a,b") var a,b,i,k int fmt.Scanf("%d %d", &a,&b) k=0 for i=a; i<=b; i++ { k++; fmt.Println(i," ") } fmt.Println("\nK=",k) } </pre>

3) Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1, 2, ..., 10 кг конфет.

PascalABC.NET		C++	
<pre> Var C: Real; i: Integer; begin Write('C: '); Readln(C); for i := 1 to 10 do Writeln(i, ' кг: ', i * C); end. </pre>		<pre> #include <iostream> using namespace std; int main() { float c; int i; cin>>c; for (i = 1; i<=10; ++i) cout<<c*i<<endl; system("pause"); return 0; } </pre>	
Python		C#	
<pre> a= int(input("->")) for i in range(1,10): print(i*a) </pre>		<pre> static void Main(string[] args){ int a, i, n, t; a=Convert.ToInt32 (Console.ReadLine ()); n=Convert.ToInt32 (Console.ReadLine ()); for (i = 1; i <= n; i++){ t = i * a; Console.Write("{0}*{1}={2}", i,a,t); } Console.WriteLine(); Console.ReadKey(); } </pre>	
Java		Go	
<pre> public static void main(String[]args){ int i; double t,s; Scanner sc=new Scanner(System.in); t=sc.nextDouble(); for(i=1; i<=10; i++){ s=i*t; System.out.print(s+ " "); } } </pre>		<pre> package main import "fmt" func main() { fmt.Println("t") var i int var s,t int fmt.Scanf("%d",&t) for i=1; i<=10; i++ { s=i* t fmt.Println(s, " ") } } </pre>	

4) Даны два целых числа A и B ($A < B$). Найти сумму всех целых чисел от A до B включительно.

PascalABC.NET		C++	
<pre> Var A, B, Sum, i: Integer; begin Write('A,B: '); Readln(A, B); sum := 0; </pre>		<pre> #include <iostream> using namespace std; int main() { int a,b, i, sum=0; cin>>a>>b; for (i =a; i<=b; ++i) sum +=i; } </pre>	

<pre> for i := A to B do Sum := Sum + i; Writeln(Sum); end. </pre>	<pre> cout<<sum<<endl; system("pause"); return 0; } </pre>
<pre> Python a= int(input("->")) b= int(input("->")) S=0; for i in range(a,b): S+=i print('S=',S) </pre>	<pre> C# static void Main(string[] args){ int a, b, i, S=0; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); for (i = a; i <= b; i++){ if (i==a) Console.Write(a); else Console.Write("{0}",i); S = S + i; } Console.Write("="+S); Console.ReadKey(); } </pre>
<pre> Java public static void main(String[] args) { int a,b,i,s=0; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); for(i=a; i<=b; i++) s=s+i; System.out.print(s+ " "); } </pre>	<pre> Go package main import "fmt" func main() { fmt.Println("a,b") var i int var s,a,b int fmt.Scanf("%d %d",&a, &b) for i=a; i<=b; i++ { s=s+i; } fmt.Println(s, " ") } } </pre>

5) Дано целое число $N (> 0)$. Найти значение выражения

$$1.1 - 1.2 + 1.3 - \dots$$

(N слагаемых, знаки чередуются). Условный оператор не использовать.

<pre> PascalABC.NET Var N, i, k: Integer; S: Real; begin Write('N: '); Readln(N); S := 0; k := -1; for i := 1 to N do begin k := k * (-1); S:=z+k*(1+0.1*i); end; Writeln(S); end. </pre>	<pre> C++ #include <cmath> #include <iostream> using namespace std; int main() { int n, i; cin>>n; float rez=0; for (i =1; i<=n; ++i) rez += pow(-1.0,(i+1))*(1.0 + (float)i/10); cout<<rez<<endl; system("pause"); return 0; } </pre>
--	--

Python	C#
<pre>n= int(input("->")) h=1; S=0; for i in range(1,n): S+=(1+0.1*i)*h h=-h print('S=',S)</pre>	<pre>static void Main(string[] args){ Double n, i, s = 0, h=1; n = Convert.ToDouble(Console.ReadLine()); for (i =1.1; i <= n; i+=0.1){ s = s + i*h; h = -h; } Console.WriteLine("\nS=" +s); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ float s=1,t=1,i,n; Scanner sc=new Scanner(System.in); n=sc.nextFloat(); for(i=(float) 1.1; i<=n; i+=0.1){ s=(s+i)*t; t=-t;} System.out.println(s); }</pre>	<pre>package main import "fmt" func main() { var i,s,t,n float64 fmt.Scanf("%v",&n) s=1 t=1 for i=1.1; i<=n; i+=0.1 { s=(s+i)*t; t=-t;} fmt.Println(s," ") }</pre>

6) Дано вещественное число A и целое число $N (> 0)$. Найти A в степени N :

$$A^N = A \cdot A \cdot \dots \cdot A$$

(числа A перемножаются N раз).

PascalABC.NET	C++
<pre>Var A, Rez: Real; N, i: Integer; begin Write('A,N: '); Readln(A, N); Rez := 1; for i := 1 to N do Rez := Rez * A; Writeln(Rez); end.</pre>	<pre>#include <iostream> using namespace std; int main() { float a, rez=1; int n,i; cin>>n>>a; for (i =1; i<=n; ++i) rez *= a; cout<<rez<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= float(input("a:")) n= int(input("n:")) S=1 for i in range(0,n): S*=a print('S=',S)</pre>	<pre>static void Main(string[] args) { Double a, n, i, p = 1; a = Convert.ToDouble(Console.ReadLine()); n = Convert.ToDouble(Console.ReadLine()); for (i = 1; i <= n; i++) p = p * a; Console.WriteLine("\np=" + p); Console.ReadKey(); }</pre>

Java	Go
<pre>public static void main(String[] args) { int i,a,n,s=1; Scanner sc=new Scanner(System.in); a=sc.nextInt(); n=sc.nextInt(); for(i=1; i<=n; i++) s=s*a; System.out.println(s); }</pre>	<pre>package main import "fmt" func main() { var s,a float64 var n,i int fmt.Scanf("%v %d", &a,&n) s=1 for i=1; i<=n; i++ { s=s*a; } fmt.Println(s," ") }</pre>

7) Дано вещественное число A и целое число $N (> 0)$. Используя один цикл, найти сумму

$$1 + A + A^2 + A^3 + \dots + A^N.$$

PascalABC.NET	C++
<pre>Var A, Rez, Pow: Real; N, i: Integer; begin Write('A,N: '); Readln(A, N); Rez := 1; Pow := 1; for i := 1 to N do begin Pow := Pow * A; Rez := Rez + Pow; end; Writeln(Rez); end.</pre>	<pre>#include <iostream> using namespace std; int main() { float a, a2=1, rez=1; int i,n; cin>>a>>n; for (i =2; i<=n; ++i) { a2*=a; rez +=a2; } cout<<rez<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>a= float(input("->")) n= int(input("->")) S=1 t=1; for i in range(0,n): t=t*a S=S+t print('S=',S)</pre>	<pre>static void Main(string[] args){ Double a, t = 1, S=1; int i, n; a= Convert.ToDouble(Console.ReadLine()); n = Convert.ToInt32(Console.ReadLine()); for (i = 1; i <= n; i++){ t = t * a; S = S + t; } Console.WriteLine(S); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[]args){ int s=1,n,i,a,t=1; Scanner sc=new Scanner(System.in); a=sc.nextInt(); n=sc.nextInt(); for(i=1; i<=n; i++){ s=s*a; } }</pre>	<pre>package main import "fmt" func main() { var s,a,t float64 var n,i int fmt.Scanf("%v %d",&a,&n) s=1</pre>

<pre> t=t+s; } System.out.println(t); } </pre>	<pre> t=0 for i=1; i<=n; i++ { s=s*a; t=t+s; } fmt.Println(t) } </pre>
--	---

8) Дано целое число $N (> 0)$. Используя один цикл, найти сумму

$$1! + 2! + 3! + \dots + N!$$

(выражение $N!$ — *N-факториал* — обозначает произведение всех целых чисел от 1 до N : $N! = 1 \cdot 2 \cdot \dots \cdot N$). Чтобы избежать целочисленного переполнения, проводить вычисления с помощью вещественных переменных и вывести результат как вещественное число.

PascalABC.NET		C++	
<pre> Var Rez: Real; N, i: Integer; begin Write(' N: '); Readln(N); Rez := 1; for i := 2 to N - 1 do begin Rez := Rez + Rez * i; end; Writeln(Rez); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int n, i; float n1=1, rez=0; cin>>n; for (i =1; i<=n; ++i){ n1 *=(float)i; rez += n1; } cout<<rez<<endl; system("pause"); return 0; } </pre>		
Python		C#	
<pre> n= int(input("->")) S=1 p=1; for i in range(1,n+1): p=p*i S=S+t print('S=',S) </pre>	<pre> static void Main(string[] args){ Double S = 0, P = 1, n, i; n=Convert.ToDouble(Console.ReadLine()); for (i = 1; i <= n; i++){ P = P * i; S = S + P; } Console.WriteLine("\nS=" + S); Console.ReadKey(); } </pre>		
Java		Go	
<pre> public static void main(String[]args){ int s=1,n,i,a,t=0; Scanner sc=new Scanner(System.in); n=sc.nextInt(); for(i=1; i<=n; i++){ s=s*i; t=t+s; } System.out.println(t); } </pre>	<pre> package main import "fmt" func main() { var t,s,n,i int fmt.Scanf("%d",&n) s=1 t=0 for i=1; i<=n; i++ { s=s*i t=t+s } fmt.Println(t) } </pre>		

9) Дано вещественное число X и целое число $N (> 0)$. Найти значение выражения

$$1 + X + X^2/(2!) + \dots + X^N/(N!)$$

($N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \exp в точке X .

PascalABC.NET		C++	
<pre> Var X, Rez, f, pow: Real; N, i: Integer; begin Write('X,N: '); Readln(X, N); Rez := 1; pow := 1; f := 1; for i := 1 to N do begin f := f * i; pow := pow * X; Rez:=Rez+pow / f; end; Writeln(Rez); end. </pre>		<pre> #include <iostream> using namespace std; int main() { float x, x1=1,n1=1,rez=1.0; int n, i; cin>>n>>x; for (i =1; i<=n; ++i){ n1 *=(float)i; x1 *= x; rez += x1/n1; } cout<<rez<<endl; system("pause"); return 0; } </pre>	
Python		C#	
<pre> from math import pow x= float(input("->")) n= int(input("->")) S=1;p=1 for i in range(1,n+1): t=pow(x,i) p=p*i S=S+t/p print('S=',S) </pre>		<pre> static void Main(string[] args){ Double x, S = 1, t; int n, P = 1, i; x = Convert.ToDouble(Console.ReadLine()); n = Convert.ToInt32(Console.ReadLine()); for (i = 1; i <= n; i++){ t = Math.Pow(x, i); P = P * i; S = S + t/ P; } Console.WriteLine("\nS=" + S); Console.ReadKey(); } </pre>	
Java		Go	
<pre> public static void main(String[]args){ int s=1,n,i,p=1,x; double t=1; Scanner sc=new Scanner(System.in); n=sc.nextInt(); x=sc.nextInt(); for(i=1; i<=n; i++){ p=p*x; s=s*i; t=t+p/s; } System.out.println(t); } </pre>		<pre> package main import "fmt" func main() { var n,t,s,p,x,i float64 fmt.Scanf("%v %v",&n,&x) s=1 p=1 t=0 for i=1; i<=n; i++ { var p=p*x var s=s*i t=t+p/s } fmt.Println(t) } </pre>	

10) Дано вещественное число X ($|X| < 1$) и целое число N (> 0). Найти значение выражения

$$X + 1 \cdot \frac{X^3}{2 \cdot 3} + 1 \cdot 3 \cdot \frac{X^5}{2 \cdot 4 \cdot 5} + \dots + 1 \cdot 3 \cdot \dots \cdot (2 \cdot N - 1) \cdot X^{2N+1} / (2 \cdot 4 \cdot \dots \cdot (2 \cdot N) \cdot (2 \cdot N + 1)).$$

Полученное число является приближенным значением функции \arcsin в точке X .

PascalABC.NET	C++
<pre> Var X, Rez, pow, T1, T2: Real; N, i: Integer; begin Write('X,N: '); Readln(X, N); Rez := X; pow := X; T1 := 1; T2 := 1; for i := 1 to N - 1 do begin T1 := T1 * (2 * i - 1); T2 := T2 * (2 * i); pow := pow * X * X; Rez := Rez + T1 * Pow / (T2 * (2 * i + 1)); end; Writeln(Rez); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ double X; int N; cin >> N>>X; double answer; double z, h; double stepen; z = stepen =1; h = X; answer = X; for(int i=1; i <= N ;i++) { // x^3 / x^5 /x^7 h = h * X * X; // stepen= 1 / 3 / 15 stepen = stepen* (2*i-1); // 6 / 40 z = z * (i*2+1)*(i*2); // (1*x^3)/6 answer += (stepen * h) / z; // 2 / 8 z = z / (i*2+1); } cout << "answer="<<answer<<endl; double y; y = asin(X); cout << "arcsin(X)= "<<y<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> from math import pow x= float(input("->")) n= int(input("->")) S=1 p=1 for i in range(1,n+1): p = p * 2*i+1; S= S+(2*i-1)*pow(x,2*i +1)/(2*i)*(2*i+1) print('S=',S) </pre>	<pre> static void Main(string[] args) { Double i=0, n, x, p = 1,s=0; n=Convert.ToDouble(Console.ReadLine()); x=Convert.ToDouble(Console.ReadLine()); for (i=1 ; i <= n; i++) { p = p * 2*i+1; s = s+(2*i-1)*Math.Pow(x,2*i+1) / (2 * i)*(2 * i + 1); } Console.Write(s); Console.ReadKey(); } </pre>

Java
<pre> public static void main(String[] args) { Double i=0, n, x, p = 1,s=0; Scanner sc=new Scanner(System.in); n=sc.nextDouble(); x=sc.nextDouble(); for (i=1 ; i <= n; i++){ p = p * 2*i+1; s=s+(2*i-1)*Math.Pow(x,2*i+1) / (2 * i)*(2 * i + 1); } System.out.println(s); } </pre>
Go
<pre> package main import ("fmt" "math") func main() { fmt.Println("x,n") var n,s,p,x,i float64 fmt.Scanf("%v %v",&n,&x) s=0 p=1 for i=1 ; i <= n; i++ { p = p * 2*i+1; s=s+(2*i-1)* math.Pow(x, 2 * i + 1) / (2 * i)*(2 * i + 1); } fmt.Println(s) } </pre>

- 11) Дано целое число $N (> 1)$ и две вещественные точки на числовой оси: A, B ($A < B$). Отрезок $[A, B]$ разбит на N равных отрезков. Вывести H — длину каждого отрезка, а также набор точек

$$A, A + H, A + 2 \cdot H, A + 3 \cdot H, \dots, B,$$

образующий разбиение отрезка $[A, B]$.

PascalABC.NET	C++
<pre> Var A, B, H: Real; N, i: Integer; begin Write('N,A,B: '); Readln(N, A, B); H := (B - A) / N; Writeln('H=: ', H); for i := 0 to N do Writeln(A + i * h); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int n,i; float a, b,h; cin>>a>>b; float h=(b-a)/n; cout<<h<<endl; for (i =0; i<=n; ++i){ cout<<a+i*h; } system("pause"); return 0; } </pre>
Python	C#
<pre> a= float(input("->")) b= float(input("->")) </pre>	<pre> static void Main(string[] args){ Double i, a, b, h, n=2; </pre>

<pre>n= int(input("->")) h=abs(b-a)/n for i in range(0,n+1): print(a+i*h)</pre>	<pre>a = Convert.ToDouble(Console.ReadLine()); b = Convert.ToDouble(Console.ReadLine()); h = Math.Abs(b - a) / n; for (i = a; i <= b;i+=h) Console.WriteLine("\ni=" + i); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { double a,h,i,n,l,b; Scanner sc=new Scanner(System.in); a=sc.nextDouble(); b=sc.nextDouble(); n=sc.nextDouble(); h=Math.abs(a-b)/n; for(i=0; i<=n; i++){ l=a+i*h; System.out.println(l); } }</pre>	<pre>package main import ("fmt" "math") func main() { fmt.Println("a,b") var a,h,i,n,l,b float64 fmt.Scanf("%v %v %v", &a,&b,&n) h=math.Abs(a-b)/n; for i=0; i<=n; i++ { l=a+i*h; fmt.Println("L=",l) } }</pre>

12) Дано целое число $N (> 0)$. Последовательность вещественных чисел A_K определяется следующим образом:

$$A_0 = 2, \quad A_K = 2 + 1/A_{K-1}, \quad K = 1, 2, \dots$$

Вывести элементы A_1, A_2, \dots, A_N .

PascalABC.NET	C++
<pre>var Ai: Real; N, i: Integer; begin Write(' N: '); Readln(N); Writeln('A(0)=2'); Ai := 2; for i := 1 to N do begin Writeln('A(',i,')=',2+1/Ai); Ai := 2 + 1 / Ai; end; end.</pre>	<pre>#include <iostream> using namespace std; int main() { int n; cin >> n; double a = 2; for (int i = 1; i <= n; ++i){ a = 2+1/a; cout << a << " "; } cout << endl; system("pause"); return 0; }</pre>
Python	C#
<pre>n= int(input("->")) a=2 print(a) for i in range(1,n+1): b=2+1/a print(b) a=b</pre>	<pre>public static void Main(string[] args) { int n; double a0 = 2, a1 = 2 + 1 / a0; n=Convert.ToInt32(Console.ReadLine()); Console.WriteLine(a0 + " " + a1); }</pre>

	<pre> for (int i = 2; i <= n; i++){ a0 = a1; a1 = 2 + 1 / a0; } Console.WriteLine(a1); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { Scanner pa=new Scanner(System.in); double a,b,n,i; n=sc.nextDouble(); a=2; System.out.println(a); for(i=1; i<=n; i++){ b=2+1/a; a=b; System.out.println(b); } } </pre>	<pre> package main import ("fmt") func main() { fmt.Println("n") var a,b,n,i float64 fmt.Scanf("%v",&n) a=2 for i=1; i<=n; i++ { b=2+1/a; a=b; } fmt.Println("b=",b) } } </pre>

Вложенные циклы

13) Даны целые положительные числа N и K . Найти сумму

$$1^k + 2^k + \dots + N^k$$

Чтобы избежать целочисленного переполнения, вычислять слагаемые этой суммы с помощью вещественной переменной и выводить результат как вещественное число.

PascalABC.NET	C++
<pre> Var N, K, i, i2: Integer; Rez, Pow: Real; begin Write('N,K: '); Readln(N, K); Rez := 0; for i := 1 to N do begin Pow := 1; for i2 := 1 to K do Pow := Pow * i; Rez := Rez + Pow; end; Writeln(Rez); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int n,k; cin>>n>>k; int i,i2; float rez=0,nk; for (i =1; i<=n; ++i) { nk=i; for (i2=1; i2<=k; ++i2) nk *=i; rez +=nk; } cout <<rez<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> n= int(input("->")) k= int(input("->")) S=0 </pre>	<pre> public static void Main(string[] args){ Double n, s=0, k, p = 1; n = Convert.ToDouble(Console.ReadLine()); } </pre>

<pre> for i in range(1,n+1): p=1 for j in range(1,k+1): p=p*i S=S+p print(S) </pre>	<pre> k = Convert.ToDouble(Console.ReadLine()); for (int i = 1; i <= n; i++){ p = 1; for (int j = 1; j <= k; j++){ p = p * i; } s = s + p; } Console.WriteLine("\ns=" + s); Console.ReadKey(true); } </pre>
Java	GO
<pre> public static void main(String[] args) { double j,n,i,k,s=0; Scanner Sc=new Scanner(System.in); n=Sc.nextDouble(); k=Sc.nextDouble(); for(i=1; i<=n; i++){ p=1; for(j=1; j<=k; j++) p=p*j; s=s+p; } System.out.println(s); } </pre>	<pre> package main import "fmt" func main() { var j,n,i,k,s,p float64 fmt.Scanf("%v %v",&n, &k) s=0 for i=1; i<=n; i++ { p=1 for j=1; j<=k; j++ { p=p*j s=s+p } fmt.Println("S=",s) } } </pre>

14) Даны целые положительные числа A и B ($A < B$). Вывести все целые числа от A до B включительно; при этом каждое число должно выводиться столько раз, каково его значение (например, число 3 выводится 3 раза).

PascalABC.NET	C++
<pre> var A, B, i, i2: Integer; begin Write('A,B: '); Readln(A, B); for i := A to B do for i2 := 1 to i do Writeln(i); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a,b; cin>>a>>b; int i,i2; for (i =a; i<=b; ++i){ for (i2=1; i2<=i; ++i2) cout<<i<<" "; cout<<endl; } system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("->")) b= int(input("->")) for i in range(a,b): for j in range(0,i): print(i) </pre>	<pre> public static void Main(string[] args) { int a,b, i,i2; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); for (i =a; i<=b; ++i){ for (i2=1; i2<=i; ++i2) </pre>

	<pre> Console.WriteLine(i); Console.WriteLine(); } Console.ReadKey(true); } </pre>
Java	GO
<pre> public static void main(String[] args) { int a,b,j,i,p; Scanner Sc=new Scanner(System.in); a=Sc.nextInt(); b=Sc.nextInt(); for(i=a; i<=b; i++){ for(j=1; j<=i; j++) System.out.print(i); } } </pre>	<pre> package main import "fmt" func main() { fmt.Println("k,n") var a,b,j,i float64 fmt.Scanf("%v %v",&a, &b) for i=a; i<=b; i++ { for j=1; j<=i; j++ { fmt.Println("i=",i) } } } </pre>

Задачи для самостоятельной работы

- 1) Даны два целых числа A и B ($A < B$). Вывести в порядке убывания все целые числа, расположенные между A и B (не включая числа A и B), а также количество N этих чисел.
- 2) Дано вещественное число — цена 1 кг конфет. Вывести стоимость 0.1, 0.2, ..., 1 кг конфет.
- 3) Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1.2, 1.4, ..., 2 кг конфет.
- 4) Даны два целых числа A и B ($A < B$). Найти произведение всех целых чисел от A до B включительно.
- 5) Даны два целых числа A и B ($A < B$). Найти сумму квадратов всех целых чисел от A до B включительно.
- 6) Дано целое число N (> 0). Найти сумму

$$1 + 1/2 + 1/3 + \dots + 1/N$$

(вещественное число).

- 7) Дано целое число N (> 0). Найти сумму

$$N^2 + (N + 1)^2 + (N + 2)^2 + \dots + (2 \cdot N)^2$$

(целое число).

- 8) Дано целое число N (> 0). Найти произведение

$$1.1 \cdot 1.2 \cdot 1.3 \cdot \dots$$

(N сомножителей).

- 9) Дано целое число N (> 0). Найти квадрат данного числа, используя для его вычисления следующую формулу:

$$N^2 = 1 + 3 + 5 + \dots + (2 \cdot N - 1).$$

После добавления к сумме каждого слагаемого выводить текущее значение суммы (в результате будут выведены квадраты всех целых чисел от 1 до N).

- 10) Дано вещественное число A и целое число N (> 0). Используя один цикл, вывести все целые степени числа A от 1 до N .
- 11) Дано вещественное число A и целое число N (> 0). Используя один цикл, найти значение выражения

$$1 - A + A^2 - A^3 + \dots + (-1)^N \cdot A^N.$$

Условный оператор не использовать.

- 12) Дано целое число N (> 0). Найти произведение

$$N! = 1 \cdot 2 \cdot \dots \cdot N$$

(N -факториал). Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и вывести его как вещественное число.

- 13) Дано целое число N (> 0). Используя один цикл, найти сумму

$$1 + 1/(1!) + 1/(2!) + 1/(3!) + \dots + 1/(N!)$$

(выражение $N!$ — N -факториал — обозначает произведение всех целых чисел от 1 до N : $N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением константы $e = \exp(1)$.

- 14) Дано вещественное число X и целое число N (> 0). Найти значение выражения

$$X - X^3/(3!) + X^5/(5!) - \dots + (-1)^N \cdot X^{2 \cdot N + 1}/((2 \cdot N + 1)!)$$

($N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \sin в точке X .

- 15) Дано вещественное число X и целое число N (> 0). Найти значение выражения

$$1 - X^2/(2!) + X^4/(4!) - \dots + (-1)^N \cdot X^{2 \cdot N}/((2 \cdot N)!)$$

($N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \cos в точке X .

- 16) Дано вещественное число X ($|X| < 1$) и целое число N (> 0). Найти значение выражения

$$X - X^2/2 + X^3/3 - \dots + (-1)^{N-1} \cdot X^N/N.$$

Полученное число является приближенным значением функции \ln в точке $1 + X$.

- 17) Дано вещественное число X ($|X| < 1$) и целое число N (> 0). Найти значение выражения

$$X - X^3/3 + X^5/5 - \dots + (-1)^N \cdot X^{2 \cdot N+1}/(2 \cdot N+1).$$

Полученное число является приближенным значением функции \arctg в точке X .

- 18) Дано вещественное число X ($|X| < 1$) и целое число N (> 0). Найти значение выражения

$$1 + X/2 - 1 \cdot X^2/(2 \cdot 4) + 1 \cdot 3 \cdot X^3/(2 \cdot 4 \cdot 6) - \dots + (-1)^{N-1} \cdot 1 \cdot 3 \cdot \dots \cdot (2 \cdot N-3) \cdot X^N/(2 \cdot 4 \cdot \dots \cdot (2 \cdot N)).$$

Полученное число является приближенным значением функции $\sqrt{1+X}$.

- 19) Дано целое число N (> 1) и две вещественные точки на числовой оси: A, B ($A < B$). Отрезок $[A, B]$ разбит на N равных отрезков. Вывести H — длину каждого отрезка, а также значения функции $F(X) = 1 - \sin(X)$ в точках, разбивающих отрезок $[A, B]$:

$$F(A), F(A + H), F(A + 2 \cdot H), \dots, F(B).$$

- 20) Дано целое число N (> 0). Последовательность вещественных чисел A_k определяется следующим образом:

$$A_0 = 1, \quad A_k = (A_{k-1} + 1)/k, \quad k = 1, 2, \dots$$

Вывести элементы A_1, A_2, \dots, A_N .

- 21) Дано целое число N (> 1). Последовательность чисел Фибоначчи F_k (целого типа) определяется следующим образом:

$$F_1 = 1, \quad F_2 = 1, \quad F_k = F_{k-2} + F_{k-1}, \quad k = 3, 4, \dots$$

Вывести элементы F_1, F_2, \dots, F_N .

- 22) Дано целое число N (> 1). Последовательность вещественных чисел A_k определяется следующим образом:

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

Вывести элементы A_1, A_2, \dots, A_N .

- 23) Дано целое число N (> 2). Последовательность целых чисел A_k определяется следующим образом:

$$A_1 = 1, \quad A_2 = 2, \quad A_3 = 3, \quad A_k = A_{k-1} + A_{k-2} - 2 \cdot A_{k-3}, \quad k = 4, 5, \dots$$

Вывести элементы A_1, A_2, \dots, A_N .

- 24) Дано целое число N (> 0). Найти сумму $1^1 + 2^2 + \dots + N^N$. Чтобы избежать целочисленного переполнения, вычислять слагаемые этой суммы с помощью вещественной переменной и выводить результат как вещественное число.

- 25) Дано целое число N (> 0). Найти сумму $1^N + 2^{N-1} + \dots + N^1$. Чтобы избежать целочисленного переполнения, вычислять слагаемые этой суммы с помощью вещественной переменной и выводить результат как вещественное число.

26) Даны целые числа A и B ($A < B$). Вывести все целые числа от A до B включительно; при этом число A должно выводиться 1 раз, число $A + 1$ должно выводиться 2 раза и т. д.

27) Для любого целого числа N больше 7, найти все такие пары чисел x и y , что $3x + 5y = N$

28) Для заданного числа x распечатать числовую последовательность:

$$\sin(x), \sin(\sin(x)), \sin(\sin(\sin(x))), \dots$$

Вычисления прекратить, когда очередной элемент последовательности станет по модулю меньше, чем 0.01.

29) Подсчитать количество сочетаний из N элементов по M ($N > M$). Для подсчета количества сочетаний используется формула:

$$C_N^M = \frac{N!}{M!(N-M)!}$$

30) Вычислить значение дроби:

$$Q = \frac{1}{1 + \frac{1}{2 + \frac{1}{3 + \frac{1}{\dots (N-2) + \frac{1}{(N-1) + \frac{1}{N}}}}}}$$

31) Для заданного натурального числа определить, образуют ли его цифры арифметическую прогрессию. Предполагается, что в числе не менее трёх цифр.

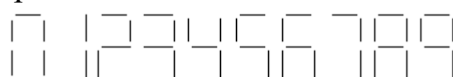
32) Получить все числа, не превышающие заданного числа N , которые делятся без остатка на все свои цифры.

33) Напечатать все натуральные четырехзначные числа, в десятичной записи которых нет одинаковых цифр, и разность двух натуральных двузначных чисел, составленных из двух последовательных первых цифр и двух последовательных последних цифр числа, равна сумме всех цифр числа.

34) В электронных часах время показывается на табло в виде последовательности цифр, указывающих часы (от 0 до 23), минуты и секунды. Например:



35) Каждая отдельная цифра на табло отображается в виде светящихся сегментов (отрезков) следующим образом:



Часы потребляют тем больше энергии, чем больше сегментов используется в записи времени. Написать программу, которая определяет время (чч.мм.сс) наибольшего и наименьшего потребления энергии часами.

- 36) Выпуклый многоугольник задан последовательностью координат своих вершин в порядке обхода: $(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_n, y_n)$. Вычислить площадь многоугольника.
- 37) В круге стоят N человек (рис.). Они пронумерованы от 1 до N . Поочередно из круга начинает выходить каждый третий человек. Это продолжается до тех пор, пока в круге не останется последний человек. Определить его номер.
Например, если в круге стояло 7 человек, то его поочередно покинут 3, 6, 2, 7, 5, 1. Оставшимся будет человек, стоявший на 4 месте.
- 38) На натуральном отрезке $[a, b]$ найдите и выведите число N с наибольшей суммой своих делителей.
- 39) Найти натуральное число из интервала (a, b) , у которого количество делителей максимально.
- 40) Найти размеры всех прямоугольников, площадь которых равна заданному натуральному числу s и стороны которых выражены натуральными числами.
- 41) Составить программу для нахождения всех натуральных решений уравнения $x^2 + y^2 = k^2$, где x, y и k лежат в интервале от 1 до 30.
- 42) Напечатать в возрастающем порядке все трехзначные числа, в десятичной записи которых нет одинаковых цифр.
- 43) Напечатать в возрастающем порядке все N -значные числа, в десятичной записи которых нет одинаковых цифр.
- 44) Составить программу нахождения цифрового корня. Цифровой корень данного числа получается, если сложить все цифры этого числа, затем все цифры найденной суммы и повторять этот процесс пока в результате будет получено однозначное число (цифра), которая и называется цифровым корнем.
- 45) Дано натуральное число n . Напечатать разложение этого числа на простые множители.
- 46) Известна зарплата каждого из N работников фирмы за каждый месяц первого квартала

Работники	Месяц		
	1	2	3
1			
2			
...			
N			

Организовать ввод информации по этой таблице и определить:

- общую сумму, выплаченную за квартал всем работникам;
- зарплату, полученную за квартал каждым работником;
- общую зарплату всех работников за каждый месяц.

47) Известны оценки каждого из N учеников по трём предметам

Работники	Месяц		
	1	2	3
1			
2			
...			
N			

Организовать ввод информации по этой таблице и определить:

- а) общее количество пятерок;
- б) количество троек у каждого ученика;
- в) количество двоек по каждому предмету.

48) Известно количество учеников в каждом из четырех классов каждой параллели школы с первой по одиннадцатую

Параллель	Класс			
	А	Б	В	Г
1				
2				
...				
11				

Организовать ввод информации по этой таблице и определить:

- а) сколько учеников учится в самом малочисленном классе школы;
- б) минимальное значение общего количество учеников, учащихся в классах одной параллели;
- в) минимальное значение общего количества учеников в классах А, Б, В и Г.

7. Цикл с условием: группа While

- 1) Даны положительные числа A и B ($A > B$). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Не используя операции умножения и деления, найти длину незанятой части отрезка A .

PascalABC.NET	C++
<pre> Var A, B, h: Integer; begin Write('A,B: '); Readln(A, B); temp := A; while temp >= 0 do h := h - B; Writeln(h + B); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a,b; cin>>a>>b; int h=a; while (h>=0){ h -= b; } cout<<h+b; system("pause"); return 0; } </pre>
Python	C#
<pre> a= int(input("->")) b= int(input("->")) while (a>=b): a=a-b print(a) </pre>	<pre> public static void Main(string[] args){ int a, b; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); while (a >= b){ a = a - b; Console.WriteLine(a + " "); } Console.WriteLine("\na=" + a); Console.ReadKey(true); } </pre>
Java	Go
<pre> public static void main(String[]args){ int a,b; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); while(a>=b){ a=a-b; } System.out.println(a); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("a,b=?") var a,b int fmt.Scanf("%d %d", &a,&b) for a>b{ a=a-b } fmt.Println(a) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { </pre>	<pre> import java.util.Scanner fun main(args: Array<String>) { </pre>

<pre> val reader = Scanner(System.`in`) var a = reader.nextInt() var b = reader.nextInt() while (a >= b) a -= b <i>print</i>(a) } </pre>	<pre> val reader = Scanner(System.`in`) var a = reader.nextInt() var b = reader.nextInt() do { a -= b }while(a>b) <i>print</i>(a) } </pre>
---	--

2) Дано целое число $N (> 0)$. Если оно является степенью числа 3, то вывести True, если не является — вывести False.

PascalABC.NET		C++	
<pre> Var A, temp: Integer; begin Write('A: '); Readln(A); temp := A; while temp < A do temp := temp*3; Write(temp = A); end. </pre>		<pre> #include <iostream> using namespace std; int main() { int a; cin>>a; while (a%3==0) a /=3; cout<<(a==1)?"True":"False"; system("pause"); return 0; } </pre>	
Python		C#	
<pre> n= int(input("->")) while (n%3==0): n/=3 print(n==1) </pre>		<pre> public static void Main(string[] args){ int n; bool s; n = Convert.ToInt32(Console.ReadLine()); while (n%3==0){ n = n / 3; } s = n == 1; Console.WriteLine(s); Console.ReadKey(true); } </pre>	
Java		GO	
<pre> public static void main(String[] args) { boolean y; int n; Scanner sc=new Scanner(System.in); n=sc.nextInt(); while(n%3==0){ n=n/3; } y=n==1; } </pre>		<pre> package main import "fmt" func main() { var n int var y bool fmt.Scanf("%v",&n) for n%3==0 { n=n/3; } y=n==1; fmt.Println("y=",y) } </pre>	

System.out.println(y);	}
}	
VB.NET	
<p>Вариант 1.</p> <pre>Sub Main() Dim a, b As Integer a = Convert.ToInt32(Console.ReadLine()) b = Convert.ToInt32(Console.ReadLine()) While a >= b a -= b End While Console.WriteLine(a) Console.ReadKey() End Sub</pre> <p>Вариант 2</p> <pre>Sub Main() Dim a, b As Integer a = Convert.ToInt32(Console.ReadLine()) b = Convert.ToInt32(Console.ReadLine()) Do While a >= b a -= b Loop Console.WriteLine(a) Console.ReadKey() End Sub</pre> <p>Вариант 3.</p> <pre>Sub Main() Dim a, b As Integer a = Convert.ToInt32(Console.ReadLine()) b = Convert.ToInt32(Console.ReadLine()) Do Until a <= b a -= b Loop Console.WriteLine(a) Console.ReadKey() End Sub</pre>	

3) Дано целое число $N (> 0)$, являющееся некоторой степенью числа 2: $N = 2^K$.
Найти целое число K — показатель этой степени.

PascalABC.NET	C++
<pre>Var N, K: Integer; begin Write(' N: '); Readln(N); K := 0; while n > 1 do begin</pre>	<pre>#include <iostream> using namespace std; int main() { int n; cin>>n; int k=0; while (n>=2){ ++k;</pre>

<pre>n := n div 2; inc(K); end; Writeln('K: ', K); end.</pre>	<pre>n/=2; } cout<<k<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>n= int(input("a: ")) p=2 k=1 while (p<n): p=p*2 k+=1 print(k)</pre>	<pre>public static void Main(string[] args) { int n, k=0; n = Convert.ToInt32(Console.ReadLine()); while (n >= 2){ n = n / 2; k++; } Console.WriteLine(k); Console.ReadKey(true); }</pre>
Java	GO
<pre>public static void main(String[] args) { int n,k=1; Scanner sc=new Scanner(System.in); n=sc.nextInt(); while(n>2){ k++; n=n/2; } System.out.println(k); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("n->!") var k,n int fmt.Scanf("%d",&n) k=1 for n>2 { k++; n=n/2; } fmt.Println(k) }</pre>

4) Дано целое число $N (> 0)$. Найти наибольшее целое число K , квадрат которого не превосходит N : $K^2 \leq N$. Функцию извлечения квадратного корня не использовать.

PascalABC.NET	C++
<pre>Var N, k: Integer; begin Write('N: '); Readln(N); K := 1; while Sqr(K) <= N do</pre>	<pre>#include <iostream> using namespace std; int main() { int n; cin>>n; int k=1; while (k*k<=n)</pre>

<pre>inc(K, 1); Writeln(K - 1); end.</pre>	<pre>++k; cout<<k-1; system("pause"); return 0; }</pre>
Python	C#
<pre>n= int(input("n:")) k=1 while (k*k<=n): k+=1 print(k-1)</pre>	<pre>public static void Main(string[] args) { Double n, k = 0, t; n = Convert.ToInt32(Console.ReadLine()); while ((k * k)<= n){ k++; } t=(k*k)>n ? k-1:k; Console.WriteLine(t); Console.ReadKey(); }</pre>
Java	GO
<pre>public static void main(String[] args) { int n,k=0,p = 0; Scanner sc=new Scanner(System.in); n=sc.nextInt(); while(k*k<=n){ k++; p=k*k; } k=p>n?k-1:k; System.out.println(k+" "); }</pre>	<pre>package main import "fmt" func main() { fmt.Println("n->") var k,n int fmt.Scanf("%d",&n) k=1 for k*k<=n { k++; } fmt.Println(k-1) }</pre>

5) Дано целое число $N (> 1)$. Вывести наибольшее из целых чисел K , для которых сумма $1 + 2 + \dots + K$ будет меньше или равна N , и саму эту сумму.

PascalABC.NET	C++
<pre>var N, K, Temp: Integer; begin Write('N: '); Readln(N); K := 0; Temp := 0; while Temp < N do begin inc(K, 1); Temp := Temp + K;</pre>	<pre>#include <iostream> using namespace std; int main() { int N; int K= 0; int S= 0; cout << " N (>1): "; cin >> N; while(S <= N) { ++K; S += K;</pre>

<pre> end; Writeln(K, ' ', Temp); end. </pre>	<pre> } cout <<"K= " << K << endl; system("pause"); return 0; } </pre>
Python	C#
<pre> n= int(input("n: ")) S=0 k=0 while (S<n): k+=1 S+=k print(k) </pre>	<pre> public static void Main(string[] args) { Double n, p=0, k=0; n = Convert.ToInt32(Console.ReadLine()); while (p < n){ k++; p= p + k; } Console.WriteLine(k); Console.ReadKey(true); } </pre>
Java	GO
<pre> public static void main(String[] args) { int n,k=0,p = 0,s=0; Scanner sc=new Scanner(System.in); n=sc.nextInt(); while(s<n){ k++; s=s+k; } System.out.println(k+" "+s); } </pre>	<pre> package main import ("fmt") func main() { fmt.Println("n") var k,n,s int fmt.Scanf("%d",&n) s=0 k=0 for s<n { k++; s=s+k; } fmt.Println(k," ",s) } </pre>

6) Начальный вклад в банке равен 1000 руб. Через каждый месяц размер вклада увеличивается на P процентов от имеющейся суммы (P — вещественное число, $0 < P < 25$). По данному P определить, через сколько месяцев размер вклада превысит 1100 руб., и вывести найденное количество месяцев K (целое число) и итоговый размер вклада S (вещественное число).

PascalABC.NET	C++
<pre> var P, S: Real; K: Integer; </pre>	<pre> #include <iostream> using namespace std; int main() { </pre>

<pre> begin Write('p: '); Readln(P); K := 1; S := 1000; while S <= 1100 do begin inc(K, 1); S := S + S * P / 100; end; Writeln(K, ' ', S); end. </pre>	<pre> float p; cin>>p; int k=1; float s=1000; while (s<=1100) { ++k; s+=s*p/100; } cout<<"K="<<k<<" S="<<s<<"\n"; system("pause"); return 0; } </pre>
Python	C#
<pre> p= float(input("->")) S=1000.0 k=1 while (S<1100): k+=1 S+=(S*p)/100; print(k) </pre>	<pre> public static void Main(string[] args) { float p; p = Convert.ToDouble(Console.ReadLine()); int k=1; float s=1000; while (s<=1100) { ++k; s+=s*p/100; } Console.WriteLine(k+" "+s); Console.ReadKey(true); } </pre>
Java	GO
<pre> public static void main(String[] args) { double p,k=0,s=1000; Scanner sc=new Scanner(System.in); p=sc.nextDouble(); while(s<=1100){ k++; s=s+1000*p/100; } System.out.println(k+" "+s); } </pre>	<pre> package main import ("fmt") func main() { fmt.Println("p") var k,p,s float64 fmt.Scanf("%v",&p) s=1000 k=0 for s<=1100 { k++; s=s+1000*p/100; } fmt.Println(k," ",s) } </pre>

7) Дано целое число $N (> 0)$. Используя операции деления нацело и взятия остатка от деления, вывести все его цифры, начиная с самой правой (разряда единиц).

PascalABC.NET	C++	
<pre> Var N: Integer; begin Write('N: '); Readln(N); while N > 0 do begin Writeln(N mod 10); N := N div 10; end; end. </pre>	<pre> #include <iostream> using namespace std; int main() { int n; cin>>n; while (n>0) { cout<<n%10; n /=10; } system("pause"); return 0; } </pre>	
Python	C#	
<pre> n= int(input("->")) while (n>0): print(n%10) n//=10 </pre>	<pre> public static void Main(string[] args) { int n; n = Convert.ToInt32(Console.ReadLine()); int k=1; while (n>0) { Console.WriteLine(n%10); n /=10; } Console.ReadKey(); } </pre>	
Java		GO
<pre> public static void main(String[] args) { int n,t,s=0; Scanner sc=new Scanner(System.in); n=sc.nextInt(); while(n!=0){ t=n%10; s=s*10+t; n=n/10; System.out.print(t+" "); } } </pre>		<pre> package main import "fmt" func main() { fmt.Println("n") var n,t,s int fmt.Scanf("%d",&n) s=0 for n!=0 { t=n%10; s=s*10+t; n=n/10; fmt.Println(t," ") } } </pre>

8) Дано целое число $N (> 0)$. С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа N цифра «2». Если имеется, то вывести True, если нет — вывести False.

PascalABC.NET		C++	
<pre> Var N: Integer; z: Boolean; begin Write('N: '); Readln(N); z := False; while (N>0) and (not z) do begin if (N mod 10) = 2 then z := True; N := N div 10; end; Writeln(z); end. </pre>		<pre> #include <iostream> using namespace std; int main() { int n; cin>>n; while((n>0)&((n%10)!=2)) { n /=10; } cout<<((n>0)&((n%10)==2)); system("pause"); return 0; } </pre>	
Python	C#		
<pre> n= int(input("->")) while ((n%10)%2==0): n//=10 print((n%10)%2==1) </pre>	<pre> public static void Main(string[] args){ int n; n = Convert.ToInt32(Console.ReadLine()); while((n>0)&((n%10)!=2)) { n /=10; } Console.WriteLine(((n>0)&(n%10)==2)); Console.ReadKey(); } </pre>		
Java		GO	
<pre> public static void main(String[] args) { int n,t; boolean y; Scanner sc=new Scanner(System.in); n=sc.nextInt(); while(n!=0){ t=n%10; n=n/10; y=(t==2); System.out.print(y+" "); } } </pre>		<pre> package main import "fmt" func main() { fmt.Println("n") var n,t int var y bool fmt.Scanf("%d",&n) for n!=0 { t=n%10; n=n/10; y=(t==2); fmt.Println(y," ") } } </pre>	

9) Даны целые положительные числа A и B . Найти их *наибольший общий делитель* (НОД), используя *алгоритм Евклида*:

$$\text{НОД}(A, B) = \text{НОД}(B, A \bmod B), \quad \text{если } B \neq 0; \quad \text{НОД}(A, 0) = A.$$

PascalABC.NET		C++	
<pre> Var A, B: Integer; begin Write('A,B: '); Readln(A, B); while (A <> 0) and (B <> 0) do begin if a > b then A := (a mod b) else B := (B mod A); end; Writeln(A + B); end. </pre>		<pre> #include <iostream> using namespace std; int main(){ int a,b; cin>>a>>b; while ((a!=0)&(b!=0)){ if (a>b) a=a%b; else b=b%a; } cout<<a+b<<endl; system("pause"); return 0; } </pre>	
Python	C#		
<pre> a= int(input("a: ")) b= int(input("b: ")) while ((a!=0) and (b!=0)): if (a>b): a=a%b else: b=b%a print(a+b) </pre>	<pre> public static void Main(string[] args){ int a,b; a = Convert.ToInt32(Console.ReadLine()); b = Convert.ToInt32(Console.ReadLine()); while ((a!=0)&(b!=0)){ if (a>b) a=a%b; else b=b%a; } Console.WriteLine(a+b); Console.ReadKey(); } </pre>		
Java		GO	
<pre> public static void main(String[] args) { int a,b; Scanner sc=new Scanner(System.in); a=sc.nextInt(); b=sc.nextInt(); while(a!=b){ if(a>b) a=a-b; else b=b-a; } System.out.println(a+ " "); } </pre>		<pre> package main import "fmt" func main() { fmt.Println("n") var a,b int fmt.Scanf("%d %d",&a,&b) for a!=b { if a>b { a=a-b; }else { b=b-a; } fmt.Println("a=",a) } } </pre>	

10) Дано целое число $N (> 1)$. Последовательность чисел Фибоначчи F_k определяется следующим образом:

$$F_1 = 1, F_2 = 1, F_k = F_{k-2} + F_{k-1}, \quad k = 3, 4, \dots$$

Проверить, является ли число N числом Фибоначчи. Если является, то вывести True, если нет — вывести False.

PascalABC.NET	C++
<pre> var N, F, F1, F2: Integer; begin Write('N: '); Readln(N); F1 := 1; F2 := 1; F := 0; while F < N do begin F := F2 + F1; F2 := F1; F1 := F; end; Writeln(F = N); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int n; cin>>n; int f1=1,f2=1,f=0; while (f<n){ f=f2+f1; f2=f1; f1=f; } cout<<(f==n)?"True":"False"; system("pause"); return 0; } </pre>
Python	C#
<pre> n= int(input("->")) f1=1 f2=1 f=0 while (f<n): f=f2+f1; f1=f2; f2=f print(f==n) </pre>	<pre> public static void Main(string[] args) { int n, f1=1,f2=1,f=0;; n=Convert.ToInt32(Console.ReadLine()); while (f<n){ f=f2+f1; f1=f2; f2=f; } Console.WriteLine(f==n?"True":"False"); Console.ReadKey(); } </pre>
Java	GO
<pre> public static void main(String[] args) { int f1,f2,f,n; boolean y=false; Scanner sc=new Scanner(System.in); n=sc.nextInt(); f1=1; f2=1; </pre>	<pre> package main import "fmt" func main() { fmt.Println("n") var f1,f2,f,n int fmt.Scanf("%d",&n) f1=1; f2=1; f=f1+f2; </pre>

<pre> f=f1+f2; while(f<=n){ f1=f2; f2=f; f=f1+f2; if(f>n) y=true; } System.out.println(f+" "); } </pre>	<pre> for f<=n { f1=f2; f2=f; f=f1+f2; fmt.Println(f," ") } } </pre>
---	---

11) Дано вещественное число $\varepsilon (> 0)$. Последовательность вещественных чисел A_k определяется следующим образом:

$$A_1 = 2, A_k = 2 + \frac{1}{A_{k-1}}, k = 2, 3, \dots$$

Найти первый из номеров K , для которых выполняется условие

$$|A_k - A_{k-1}| < \varepsilon,$$

и вывести этот номер, а также числа A_{k-1} и A_k .

PascalABC.NET		C++	
<pre> Var k: Integer; Ak1, Ak, e: Real; begin Write('e: '); Readln(e); Ak1 := 0; Ak := 2; K := 1; while Abs(Ak-Ak1)>=e do begin Inc(k, 1); Ak1 := Ak; Ak := 2 + 1 / Ak1; end; Writeln(k, ' ', Ak1, ' ', Ak); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ float e; cin>>e; float ak1=0,ak=2; int k=1; while (fabs(ak-ak1)>=e) { ++k; ak1=ak; ak=2+1/ak1; } cout<<k<<ak1<<ak<<endl; system("pause"); return 0; } </pre>	<pre> Python e= float(input("e:")) k=2 a1=2 a2=2+1.0/a1; h=abs(a2-a1); while (h>=e): k+=1 a1=a2; </pre>	<pre> C# public static void Main(string[] args){ double e, ak1=0,ak=2; e= Convert.ToSingle(Console.ReadLine()); int k=1; while (fabs(ak-ak1)>=e) { ++k; ak1=ak; ak=2+1/ak1; } } </pre>

<pre> a2=2+1.0/a1; h=abs(a2-a1); print(k, ' ',a1, ' ',a2) </pre>	<pre> } Console.WriteLine(k+" "+"ak1+" "+"ak); Console.ReadKey(); } </pre>
Java	GO
<pre> public static void main(String[] args) { double a1,a2, h,eps; int k=2; Scanner sc=new Scanner(System.in); eps=sc.nextDouble(); a1=2; a2=2+1.0/a1; h=Math.abs(a2-a1); while (h>=eps){ k++; a1=a2; a2=2+1.0/a1; h=Math.abs(a2-a1); } System.out.println("k="+k+" a1="+ a1+" a2="+a2); } </pre>	<pre> package main import ("fmt" "math") func main() { var a1,a2, h,eps float64 var k int fmt.Scanf("%v",&eps) a1=2; k=2; a2=2+1.0/a1; h=math.Abs(a2-a1); for h>=eps { k++; a1=a2; a2=2+1.0/a1; h=math.Abs(a2-a1); } fmt.Println("k=",k,"\na1=", a1, "\na2=",a2) } </pre>

12) Даны положительные числа A , B , C . На прямоугольнике размера $A \neq B$ размещено максимально возможное количество квадратов со стороной C (без наложений). Найти количество квадратов, размещенных на прямоугольнике. Операции умножения и деления не использовать.

PascalABC.NET	C++
<pre> Var A, B,H, C: Real; K: integer; begin Readln(A, B, C); K := 0; while (A - C >= 0) do begin A := A - C; H := B; while (H - C >= 0) do begin H := H - C; inc(K, 1); end; end; Writeln(k); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ float a,b,c; cin>>a>>b>>c; int k=0; float temp; while ((a-c)>=0) { a-=c; temp=b; while (temp-c>=0) { temp-=c; ++k; } } cout<<k<<endl; system("pause"); return 0; } </pre>

Python	C#
<pre> a= float(input("a: ")) b= float(input("b: ")) c= float(input("c: ")) k=0 h=0 while ((a-c)>=0): a-=c h=b while (h-c>=0): h-=c k+=1 print(k) </pre>	<pre> public static void Main(string[] args) { float a,b,c; a=Convert.ToSingle(Console.ReadLine()); b=Convert.ToSingle(Console.ReadLine()); c=Convert.ToSingle(Console.ReadLine()); int k=0; float temp; while ((a-c)>=0) { a-=c; temp=b; while (temp-c>=0){ temp-=c; ++k; } } Console.WriteLine(k); Console.ReadKey(); } </pre>
Java	GO
<pre> public static void main(String[]args) { double A,B,C,h=0,k=0,s=0,i=1; Scanner sc=new Scanner(System.in); A=sc.nextDouble(); B=sc.nextDouble(); C=sc.nextDouble(); while(A>=C){ A=A-C; k++;} while(B>=C){ B=B-C; h++; } while(i<=k){ s=s+h; i++; } System.out.println(s+" "); } </pre>	<pre> package main import "fmt" func main() { fmt.Println("n") var A,B,C,h,k,s,i float64 fmt.Scanf("%v %v %v", &A,&B,&C) h=0; k=0; s=0; i=1; for A>=C { A=A-C; k++; } for B>=C { B=B-C; h++; } for i<=k { s=s+h; i++; } fmt.Println(s," ") } </pre>

Задачи для самостоятельной работы

- 1) Даны положительные числа A и B ($A > B$). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Не используя операции умножения и деления, найти количество отрезков B , размещенных на отрезке A .
- 2) Даны целые положительные числа N и K . Используя только операции сложения и вычитания, найти частное от деления нацело N на K , а также остаток от этого деления.
- 3) Дано целое число N (> 0). Найти *двойной факториал* N :
$$N!! = N \cdot (N-2) \cdot (N-4) \cdot \dots$$
(последний сомножитель равен 2, если N — четное, и 1, если N — нечетное). Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и вывести его как вещественное число.
- 4) Дано целое число N (> 0). Найти наименьшее целое положительное число K , квадрат которого превосходит N : $K^2 > N$. Функцию извлечения квадратного корня не использовать.
- 5) Дано целое число N (> 1). Найти наименьшее целое число K , при котором выполняется неравенство $3^K > N$.
- 6) Дано целое число N (> 1). Найти наибольшее целое число K , при котором выполняется неравенство $3^K < N$.
- 7) Дано целое число N (> 1). Вывести наименьшее из целых чисел K , для которых сумма $1 + 2 + \dots + K$ будет больше или равна N , и саму эту сумму.
- 8) Дано число A (> 1). Вывести наименьшее из целых чисел K , для которых сумма $1 + 1/2 + \dots + 1/K$ будет больше A , и саму эту сумму.
- 9) Дано число A (> 1). Вывести наибольшее из целых чисел K , для которых сумма $1 + 1/2 + \dots + 1/K$ будет меньше A , и саму эту сумму.
- 10) Спортсмен-лыжник начал тренировки, пробежав в первый день 10 км. Каждый следующий день он увеличивал длину пробега на P процентов от пробега предыдущего дня (P — вещественное, $0 < P < 50$). По данному P определить, после какого дня суммарный пробег лыжника за все дни превысит 200 км, и вывести найденное количество дней K (целое) и суммарный пробег S (вещественное число).
- 11) Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.
- 12) Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти число, полученное при прочтении числа N справа налево.
- 13) Дано целое число N (> 0). С помощью операций деления нацело и взятия остатка от деления определить, имеются ли в записи числа N нечетные цифры. Если имеются, то вывести True, если нет — вывести False.
- 14) Дано целое число N (> 1). Если оно является *простым*, то есть не имеет положительных делителей, кроме 1 и самого себя, то вывести True, иначе вывести False.
- 15) Дано целое число N (> 1). Найти первое число Фибоначчи, большее N .
- 16) Дано целое число N (> 1), являющееся числом Фибоначчи: $N = F_K$. Найти целые числа F_{K-1} и F_{K+1} — предыдущее и последующее числа Фибоначчи.
- 17) Дано целое число N (> 1), являющееся числом Фибоначчи: $N = F_K$. Найти целое число K — порядковый номер числа Фибоначчи N .

18) Дано вещественное число $\varepsilon (> 0)$. Последовательность вещественных чисел A_K определяется следующим образом:

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

Найти первый из номеров K , для которых выполняется условие

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

и вывести этот номер, а также числа A_{K-1} и A_K .

19) Организовать непрерывный ввод чисел с клавиатуры, пока пользователь не введёт 0. После ввода нуля, показать на экран количество чисел, которые были введены, их общую сумму и среднее арифметическое.

20) Дано число a ($1 < a \leq 1,5$). Из чисел $1 + \frac{1}{2}, 1 + \frac{1}{3}, \dots$ напечатать те, которые не меньше a .

21) Для заданного числа x распечатать числовую последовательность:

$$\sin(x), \sin(\sin(x)), \sin(\sin(\sin(x))), \dots$$

Вычисления прекратить, когда очередной элемент последовательности станет по модулю меньше, чем 0.01.

22) Даны положительные вещественные числа a, x, ε . В последовательности y_1, y_2, \dots , образованной по закону:

$$23) y_i = \frac{1}{2} \left(y_{i-1} + \frac{x}{y_{i-1}-1} \right), \quad i = 1, 2, \dots,$$

24) найти первый член y_n , для которого выполнено неравенство $|y_n^2 - y_{n-1}^2| < \varepsilon$.

25) Используя метод деления отрезка пополам, найти приближенное (с точностью 0,001) значение корня уравнения $f(x) = 0$ на отрезке $[a, b]$:

$$26) f(x) = x^4 + 2x^3 - x - 1, \quad a = 0, \quad b = 1;$$

$$27) f(x) = x^3 - 0,2x^2 - 0,2x - 1,2, \quad a = 1, \quad b = 1,5.$$

28) Составить таблицу значений функции $y = 5 - x^2/2$ на отрезке $[-5; 5]$ с шагом 0.5.

29) Из натурального числа удалить заданную цифру. Число и цифру вводить с клавиатуры.

30) Составить программу разложения натурального числа n на простые множители.

31) Вычислить сумму элементов следующей последовательности с точностью 0,001:

$$S = 1 + \frac{2}{3 \cdot 3} + \frac{4}{5 \cdot 3^2} + \frac{6}{7 \cdot 3^3} + \frac{8}{9 \cdot 3^4} + \dots$$

32) Вычислять с использованием цикла *while* квадратные корни из чисел 900, 893, 886, 879 и т.д. до тех пор, пока это можно делать.

33) Написать программу, которая выводит на экран значение функции

$$y = 5x^2 - 3x - 10$$

в диапазоне от -4 до 4 , с шагом изменения аргумента в 0.5 .

34) Написать программу, вычисляющую среднее арифметическое N чисел, введенных с клавиатуры.

- 35) Определите среднее значение всех элементов последовательности, завершающейся числом 0.
- 36) Последовательность состоит из натуральных чисел и завершается числом 0. Определите, сколько элементов этой последовательности равны ее наибольшему элементу.
- 37) Дана последовательность натуральных чисел, завершающаяся числом 0. Определите, какое наибольшее число подряд идущих элементов этой последовательности равны друг другу.
- 38) Дана последовательность натуральных чисел x_1, x_2, \dots, x_n . *Стандартным отклонением* называется величина

$$\sigma = \sqrt{\frac{(x_1 - s)^2 + (x_2 - s)^2 + \dots + (x_n - s)^2}{n - 1}},$$

где $s = (x_1 + x_2 + \dots + x_n)/n$ - среднее арифметическое последовательности.

8. Последовательности: группа Series

Во всех заданиях данной группы предполагается, что исходный набор содержит ненулевое число элементов (в частности, число N всегда больше нуля). В заданиях на обработку нескольких наборов чисел количество наборов K также всегда является ненулевым.

Для решения заданий из данной группы следует использовать «однопроходные» алгоритмы, позволяющие получить требуемый результат после *однократного* просмотра набора исходных данных. Однопроходные алгоритмы обладают важным преимуществом: для них не требуется хранить в памяти одновременно весь набор данных, поэтому при программной реализации этих алгоритмов *можно не использовать массивы*.

1) Даны десять вещественных чисел. Найти их сумму.

PascalABC.NET		C++	
<pre> Var R, Rez: Real; i: Integer; begin Rez := 0; for i := 1 to 10 do begin Readln(R); Rez := Rez + R; end; Writeln(Rez); end. </pre>		<pre> int main() { float S = 0, r; for (int i = 1; i <= 10; ++i) { cin >> r; S+= r; } cout << S; return 0; } </pre>	
Python		C#	
<pre> S=0 for i in range(1,10): a=float(input("->")) S=S+a print(S) </pre>		<pre> static void Main(string[] args) { float a, S = 0; for (int i=0; i<10; i++){ a=Convert.ToSingle(Console.ReadLine()); S = S + a; } Console.WriteLine(S); Console.ReadKey(); } </pre>	
Java		Go	
<pre> public static void main(String[] args){ Scanner sc=new Scanner(System.in); float a, S = 0; for (int i=0; i<10; i++){ a =sc.nextFloat(); S+=a; } System.out.println(S); } </pre>		<pre> package main import "fmt" func main() { var S=0.0 var a=0.0 for i:=1; i<=10; i++){ fmt.Scan(&a) S=S+a } fmt.Println(S) } </pre>	

Kotlin
<pre>import java.util.Scanner fun main(args: Array<String>){ var S: Double = 0.0 val reader = Scanner(System.`in`) for(i in 1..10){ val x = reader.nextDouble() S=S+x; } println(S) }</pre>
VB.NET
<pre>Sub Main() Dim a, S As Single For i As Integer = 1 To 10 a = Convert.ToSingle(Console.ReadLine()) S += a Next Console.WriteLine(S) Console.ReadKey() End Sub</pre>

- 2) Дано целое число N и набор из N положительных вещественных чисел. Вывести в том же порядке целые части всех чисел из данного набора (как вещественные числа с нулевой дробной частью), а также сумму всех целых частей.

PascalABC.NET	C++
<pre>Begin write('N = '); var N := readInteger; var S: real:= 0; for var i := 1 to N do begin var a := readReal; S := S + int(a); write(int(a), ' ') end; writeln; writeln(S); end.</pre>	<pre>int main() { int S = 0, i; float r; for (i = 1; i <= 10; ++i) { cin >> r; S+= (int)r; cout << (int)r << endl; } cout << S; return 0; }</pre>
Python	
<pre>S=0 n=int(input("->")) for i in range(0,n): a=float(input("->")) S=S+int(a) print(int(a)) print(S)</pre>	

- 3) Даны целые числа K, N и набор из N целых чисел. Если в наборе имеются числа, меньшие K , то вывести True; в противном случае вывести False.

PascalABC.NET	C++
<pre> var check:Boolean; i,N,K,R:Integer; begin (N,K):=ReadInteger2; check:=False; for i:=1 to N do begin Readln(R); if R<K then check:=True; end; Writeln(check); End </pre>	<pre> int main() { int k, n, i, r; bool b = 0; cin >> n >> k; for (i = 1; i <= n; i++) { cin >> r; if (r<k) b = 1; } cout<<b<<endl; system("pause"); return 0; } </pre>

4) Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество чисел в наборе, меньших K .

PascalABC.NET	C++
<pre> var i, R, K: Longint; begin readln(K); R := 1; i := 0; while(R <> 0) do begin Readln(R); if R < K then i += 1; end; i -= 1; Writeln(i); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int a, i = -1, k; bool y = 0; cin >> k; do { i++; cin >> a; if (a > k) y = 1; } while (a != 0); cout<<y<<" "<<i<<endl; system("pause"); return 0; } </pre>

5) Дано вещественное число B , целое число N и набор из N вещественных чисел, упорядоченных по возрастанию. Вывести элементы набора вместе с числом B , сохраняя упорядоченность выводимых чисел.

PascalABC.NET	C++
<pre> var n, i, c: integer; b, num: real; begin read(n); read(b); c := 0; for i := 1 to n do </pre>	<pre> int main(){ int n; float b,r=1.0; cout<<"N, B"; cin >> n >> b; bool f=1; for (int i=1; i<=n; i++){ cin>>r; if ((b<=r)&&f){ </pre>

<pre> begin readln(num); if (c=0) and (b<num) then begin writeln(b); c := 1; end; writeln(num); end; if c = 0 then write(b); end.</pre>	<pre> cout<<b; f=0; } cout<< r; system("pause"); return 0; }</pre>
--	--

6) Дано целое число $N (> 1)$ и набор из N целых чисел. Вывести те элементы в наборе, которые меньше своего левого соседа, и количество K таких элементов.

PascalABC.NET	C++
<pre> var i, K, N, Num, Num1: Integer; begin Readln(N); K := 0; for i := 1 to N do begin Readln(Num); if (Num1>Num) and (i<>1) then begin writeln(Num); inc(K); end; Num1 := Num; end; Writeln(K); end.</pre>	<pre> int main(){ int n, i, a,b, kol=0; cin>>n>>b; for (i=1; i<=n; i++){ cin >> a; if (a < b) { cout << a << "\n"; kol++; } b = a; } cout << kol << endl; system("pause"); return 0; }</pre>

7) Дано целое число $N (> 1)$ и набор из N вещественных чисел. Проверить, образует ли данный набор возрастающую последовательность. Если образует, то вывести True, если нет — вывести False.

PascalABC.NET	C++
<pre> Var a, Prev: real; N, i: byte; IncSeq: boolean; begin write('N = '); readln(N); IncSeq := true; for i := 1 to N do begin read(a); if i > 1 then</pre>	<pre> #include <iostream> using namespace std; int main(){ int n, i, a,b; bool y = 1; cin>>n>>b; for (i=2; i<=n; i++){ cin >> a; if (b>a) y = 0; b = a; }</pre>

<pre> if IncSeq then if not (a > Prev) then IncSeq := false; Prev := a end; writeln; writeln(IncSeq); end. </pre>	<pre> cout << y << endl; system("pause"); return 0; } </pre>
--	--

8) Дано целое число $N (> 2)$ и набор из N вещественных чисел. Набор называется *пилообразным*, если каждый его внутренний элемент либо больше, либо меньше обоих своих соседей (то есть является «зубцом»). Если данный набор является пилообразным, то вывести 0; в противном случае вывести номер первого элемента, не являющегося зубцом.

PascalABC.NET	
<pre> Var a, Left, Right: real; N, i, Number: byte; begin write('N = '); readln(N); Number := 0; for i := 1 to N do begin read(Right); if i = 1 then Left := Right else if i = 2 then a := Right else begin if Number = 0 then if not (((a < Right) and (a < Left)) or ((a > Right) and (a > Left))) then Number := i; Left := a; a := Right end end; writeln; if Number = 0 then writeln('Последовательность пилообразная') else writeln('Номер элемента, нарушающего закономерность:', Number) end. </pre>	
C++	
<pre> #include <iostream> using namespace std; bool pilob(int a, int b, int c) { return ((a > b) && (b < c)) ((a < b) && (b > c)); } </pre>	

```

}
int main(){
    int n, i, a, b, h;
    bool y= 1;
    cin>>n>>a>>h;
    for (i=3; i<=n; i++){
        cin >> b;
        if (!pilob(a, h, b))
            y=0;
        cout << a << " " << h << " " << b << endl;
        a = h;
        h = b;
    }
    cout << y << endl;
    system("pause");
    return 0;
}

```

Вложенные циклы

9) Даны целые числа K, N и набор из N вещественных чисел: A_1, A_2, \dots, A_N . Вывести K -е степени чисел из данного набора:

$(A_1)^K, (A_2)^K, \dots, (A_N)^K$.

PascalABC.NET
<pre> var K, N, i, j: byte; A, S: real; begin write('K = '); readln(K); write('N = '); readln(N); for i := 1 to N do begin writeln; write('A', i, ' = '); readln(A); S := 1; for j := 1 to K do S := S * A; writeln(' --> ', A, '^', K, ' = ', S) end; end. </pre>
C++
<pre> int main(){ int k, n; cin >> n >> k; int i, j; float a, P; for (i = 1; i <= n; ++i) { </pre>

```

    cin >> a;
    P = 1;
    for (j = 1; j <= k; j++)
        P *= a;
    cout << P << endl;
}
system("pause");
return 0;
}

```

10) Даны целые числа K , N , а также K наборов целых чисел по N элементов в каждом наборе. Вывести общую сумму всех элементов, входящих в данные наборы.

PascalABC.NET
<pre> var K, N, Ni, Ki, A, S: Integer; begin Write('N: '); Readln(N); Write('K: '); Readln(K); S := 0; for Ki := 1 to K do for Ni := 1 to N do begin write('Введите ', Ni, ' элемент из ', Ki, ' набора :'); Readln(A); S := S + A; end; writeln('Сумма элементов равна: ', S); end. </pre>
C++
<pre> #include <iostream> using namespace std; int main(){ int k, n, i, j, a, S=0; cin >> k >> n; for (i = 1; i <= k; ++i) for (j = 1; j <= n; j++){ cin >> a; S += a; } cout<<S<<endl; system("pause"); return 0; } </pre>

11) Даны целые числа K , N , а также K наборов целых чисел по N элементов в каждом наборе. Для каждого набора вывести сумму его элементов.

PascalABC.NET	C++
<pre> var a, Sum: integer; K, N, i, j: byte; begin write('K = '); readln(K); write('N = '); readln(N); for i := 1 to K do begin writeln; Sum := 0; for j := 1 to N do begin read(a); Sum := Sum + a end; writeln(' S= ', Sum) end; end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int k, n, i, j,a, S=0; cin >> k >> n; for (i = 1; i <= k; ++i) { S = 0; for (j = 1; j <= n; j++) { cin >> a; S += a; } cout << "S="<<S << endl; } system("pause"); return 0; } </pre>

13) Дано целое число K , а также K наборов ненулевых целых чисел. Признаком завершения каждого набора является число 0. Для каждого набора вывести количество его элементов. Вывести также общее количество элементов во всех наборах.

PascalABC.NET	C++
<pre> var a, Q, count: integer; K, i: byte; begin write('K = '); readln(K); Q := 0; for i := 1 to K do begin count := 0; repeat read(a); inc(count) until a = 0; writeln('--> ', count); Q := Q + count end; writeln('Q= ', Q); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int n, i, j,a, S=0, kol; cin >> n; for (i = 1; i <= n; ++i) { kol = 0; cin >> a; while (a!=0){ kol++; cin >> a; } cout << "kol=" << kol << endl; S += kol; } cout << S << endl; system("pause"); return 0; } </pre>

14) Дано целое число K , а также K наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Для каждого набора выполнить следующее действие: если элементы набора

возрастают, то вывести 1; если элементы набора убывают, то вывести -1; если элементы набора не возрастают и не убывают, то вывести 0.

PascalABC.NET	C++
<pre> var a, b, K, i, p: integer; begin readln(K); for i := 1 to K do begin read(b); read(a); if a < b then p := -1 else if a > b then p := 1 else p := 0; while a <> 0 do begin if p <> 0 then if (p = -1) and (a < b) or (p=1) and (a>b) then b := a else p := 0; read(a) end; writeln(p) end; end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int k; cin >> k; int count=0,i,a=1,b; for (i=1; i<=k; ++i){ cout<<">"; cin>>a; int checkup=0, checkdown=0; while(a!=0){ b=a; cout<<">"; cin>>a; if ((b<=a)&&(a!=0)) checkup=1; if ((b>=a)&&(a!=0)) checkdown=1; } if (checkup&checkdown) cout << "0" << endl; else if (checkup) cout << "1" << endl; else if (checkdown) cout << "-1" << endl; } system("pause"); return 0; } </pre>

Задачи для самостоятельной работы

- 1) Даны десять вещественных чисел. Найти их произведение.
- 2) Даны десять вещественных чисел. Найти их среднее арифметическое.
- 3) Дано целое число N и набор из N вещественных чисел. Вывести сумму и произведение чисел из данного набора.
- 4) Дано целое число N и набор из N положительных вещественных чисел. Вывести в том же порядке дробные части всех чисел из данного набора (как вещественные числа с нулевой целой частью), а также произведение всех дробных частей.
- 5) Дано целое число N и набор из N вещественных чисел. Вывести в том же порядке округленные значения всех чисел из данного набора (как целые числа), а также сумму всех округленных значений.
- 6) Дано целое число N и набор из N целых чисел. Вывести в том же порядке все четные числа из данного набора и количество K таких чисел.
- 7) Дано целое число N и набор из N целых чисел. Вывести в том же порядке номера всех нечетных чисел из данного набора и количество K таких чисел.
- 8) Дано целое число N и набор из N целых чисел. Если в наборе имеются положительные числа, то вывести True; в противном случае вывести False.
- 9) Дан набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество чисел в наборе.
- 10) Дан набор ненулевых целых чисел; признак его завершения — число 0. Вывести сумму всех положительных четных чисел из данного набора. Если требуемые числа в наборе отсутствуют, то вывести 0.
- 11) Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер первого числа в наборе, большего K . Если таких чисел нет, то вывести 0.
- 12) Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер последнего числа в наборе, большего K . Если таких чисел нет, то вывести 0.
- 13) Дано целое число N и набор из N целых чисел, упорядоченный по возрастанию. Данный набор может содержать одинаковые элементы. Вывести в том же порядке все различные элементы данного набора.
- 14) Дано целое число $N (> 1)$ и набор из N целых чисел. Вывести те элементы в наборе, которые меньше своего правого соседа, и количество K таких элементов.
- 15) Дано целое число $N (> 1)$ и набор из N вещественных чисел. Если данный набор образует убывающую последовательность, то вывести 0; в противном случае вывести номер первого числа, нарушающего закономерность.
- 16) Дано целое число N и набор из N целых чисел, содержащий по крайней мере два нуля. Вывести сумму чисел из данного набора, расположенных между последними двумя нулями (если последние нули идут подряд, то вывести 0).
- 17) Дано целое число N и набор из N целых чисел, содержащий по крайней мере два нуля. Вывести сумму чисел из данного набора, расположенных между первым и последним нулем (если первый и последний нули идут подряд, то вывести 0).

Вложенные циклы

18) Дано целое число N и набор из N вещественных чисел: A_1, A_2, \dots, A_N . Вывести следующие числа:

$$A_1, (A_2)^2, \dots, (A_{N-1})^{N-1}, (A_N)^N.$$

19) Дано целое число N и набор из N вещественных чисел: A_1, A_2, \dots, A_N . Вывести следующие числа:

$$(A_1)^N, (A_2)^{N-1}, \dots, (A_{N-1})^2, A_N.$$

20) Даны целые числа K, N , а также K наборов целых чисел по N элементов в каждом наборе. Найти количество наборов, содержащих число 2. Если таких наборов нет, то вывести 0.

21) Даны целые числа K, N , а также K наборов целых чисел по N элементов в каждом наборе. Для каждого набора вывести номер его первого элемента, равного 2, или число 0, если в данном наборе нет двоек.

22) Даны целые числа K, N , а также K наборов целых чисел по N элементов в каждом наборе. Для каждого набора вывести номер его последнего элемента, равного 2, или число 0, если в данном наборе нет двоек.

23) Даны целые числа K, N , а также K наборов целых чисел по N элементов в каждом наборе. Для каждого набора выполнить следующее действие: если в наборе содержится число 2, то вывести сумму его элементов; если в наборе нет двоек, то вывести 0.

24) Дано целое число K , а также K наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Найти количество наборов, элементы которых возрастают.

25) Дано целое число K , а также K наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Найти количество наборов, элементы которых возрастают или убывают.

26) Дано целое число K , а также K наборов ненулевых целых чисел. Каждый набор содержит не менее трех элементов, признаком его завершения является число 0. Найти количество пилообразных наборов.

27) Дано целое число K , а также K наборов ненулевых целых чисел. Каждый набор содержит не менее трех элементов, признаком его завершения является число 0. Для каждого набора выполнить следующее действие: если набор является пилообразным, то вывести количество его элементов; в противном случае вывести номер первого элемента, который не является зубцом.

9. Процедуры и функции: группа Proc

1. Процедуры с числовыми параметрами

1) Описать процедуру PowerA3(A, B), вычисляющую третью степень числа A и возвращающую ее в переменной B (A — входной, B — выходной параметр; оба параметра являются вещественными). С помощью этой процедуры найти третьи степени пяти данных чисел.

PascalABC.NET		C++	
<pre> procedure PowerA3(A: Real; var B: Real); begin B := A * A * A; end; var i: Integer; A, B: Real; begin for i := 1 to 5 do begin Write('A: '); Readln(A); PowerA3(A, B); Writeln(B); end; end; end. </pre>		<pre> void powera3(float a, float &b){ b=a*a*a; } int main(){ int i; for(i=1;i<=5;++i){ float a,b; cin>>a>>b powera3(a, b); cout<<b<<" "; } return 0; } </pre>	
Python		C#	
<pre> def powerA3(x): return a*a*a for i in range(0,5): a=float(input("a")) print(powerA3(a)) </pre>		<pre> static void powerA3(float a, out float b){ b = a * a * a; } static void Main(string[] args){ float a, b; for (int i=0; i<5; i++){ a=Convert.ToSingle(Console.ReadLine()); powerA3(a, out b); Console.WriteLine(b); } Console.ReadKey(); } </pre>	
Java		Go	
<pre> static void PowerA3(double a){ double b=a*a*a; System.out.println(b); } public static void main(String[] args){ Scanner sc=new Scanner(System.in); double a, S = 0; for (int i=0; i<5; i++){ a=sc.nextDouble(); PowerA3(a); } } </pre>		<pre> package main import "fmt" func PowerA3(a float32) { var b=a*a*a; fmt.Println(b) } func main() { var a float32 for i:=0; i<5; i++ { fmt.Scan(&a) PowerA3(a) } } </pre>	

Kotlin
<pre>import java.util.Scanner fun powerA3(a: Double) { var b=Math.pow(a,3.0) println(b) } fun main(args: Array<String>){ val reader = Scanner(System.`in`) for(i in 1..5){ val x = reader.nextDouble() powerA3(x); } }</pre>
VB.NET
<pre>Sub PowerA3(ByRef x As Integer, ByVal a As Integer) x = a * a * a End Sub Sub Main() Dim a, b As Integer For i As Integer = 1 To 5 a = Convert.ToInt32(Console.ReadLine()) PowerA3(b, a) Console.WriteLine(b) Next Console.ReadKey() End Sub</pre>

2) Описать процедуру $Mean(X, Y, a, g)$, вычисляющую *среднее арифметическое* $a = (X + Y)/2$ и *среднее геометрическое* $g = \sqrt{X \cdot Y}$ двух положительных чисел X и Y (X и Y — входные, a и g — выходные параметры вещественного типа). С помощью этой процедуры найти среднее арифметическое и среднее геометрическое для пар (A, B) , (A, C) , (A, D) , если даны A, B, C, D .

PascalABC.NET	C++
<pre>procedure Mean(X, Y: Real; var AMean, GMean: Real); begin AMean := (X + Y) / 2; GMean := sqrt(X * Y); end; var i: Integer; A, B, C, D, AMean, GMean: Real; begin Write('A,B,C,D: '); Readln(A, B, C, D); Mean(A, B, AMean, GMean); Writeln(AMean, ' ', GMean); Mean(A, C, AMean, GMean);</pre>	<pre>#include <iostream> #include <cmath> using namespace std; void Mean(float x, float y, float &a, float &g) { a = (x + y) / 2; g = sqrt(x*y); } int main(){ float a, b, c, d, x, y; cin >> a >> b >> c >> d; Mean(a, b, x, y); cout << x << " " << y << endl; Mean(a, c, x, y); cout << x << " " << y << endl; Mean(a, d, x, y); cout << x << " " << y << endl; system("pause"); return 0; }</pre>

<pre> Writeln(AMean, ' ', GMean); Mean(A, D, AMean, GMean); Writeln(AMean, ' ', GMean); end. </pre>	
Python	C#
<pre> import math def Mean(x,y): a=(x+y)/2 g=math.sqrt(x*y) return(a,g) for i in range(0,5): a=float(input("a")) b=float(input("b")) print(Mean(a,b)) </pre>	<pre> static void Mean(double x, double y, out double a, out double g) { a=(x+y)/2; g=Math.Sqrt(x*y); } static void Main(string[] args) { double a,b,c,d,x,y; a=Convert.ToSingle(Console.ReadLine()); b=Convert.ToSingle(Console.ReadLine()); c=Convert.ToSingle(Console.ReadLine()); d=Convert.ToSingle(Console.ReadLine()); Mean(a,b, out x, out y); Console.WriteLine("AMean={0} GMean={1}", x,y); Mean(a, c, out x, out y); Console.WriteLine("AMean={0} GMean={1}", x, y); Mean(a, d, out x, out y); Console.WriteLine("AMean={0} GMean={1}", x, y); Console.ReadKey(); } </pre>
Go	
<pre> package main import ("fmt" "math") func Mean(x,y float64) (float64,float64) { var a=(x+y)/2; var g=math.Sqrt(x*y) return a,g } func main() { var a,b float64 for i:=0; i<5; i++ { fmt.Print("a, b->") fmt.Scan(&a,&b) fmt.Println(Mean(a,b)) } } </pre>	

3) Описать процедуру DigitCountSum(K , C , S), находящую количество C цифр целого положительного числа K , а также их сумму S (K — входной, C и S — выходные параметры целого типа). С помощью этой процедуры найти количество и сумму цифр для каждого из пяти данных целых чисел.

PascalABC.NET	C++
<pre> procedure DigitCountSum(K: Integer; var C, S: Integer); var TempI: Integer; begin C := 0; S := 0; while (k <> 0) do begin S := S + (K mod 10); C := C + 1; K := K div 10; end; end; var i, K, C, S: Integer; begin for i := 1 to 5 do begin Write('K: '); Readln(K); DigitCountSum(K, C, S); Writeln(C, ' ', S); end; end. </pre>	<pre> #include <iostream> using namespace std; void DigitCountSum(int k, int &c, int &S) { c = S = 0; while (k != 0) { S = S + k / 10; c++; k = k / 10; } } int main(){ int i,k, c, S; for (i = 0; i < 5; i++) { cin >> k; DigitCountSum(k, c, S); cout << "c="<<c << " S=" << S << endl; } system("pause"); return 0; } </pre>
Go	
<pre> package main import "fmt" func digitcountsum(x int) (int, int) { var s=0 var k=0 for x!=0{ s=s+x%10 x=x/10 k++; } return k,s } func main() { var a int for i:=0; i<5; i++ { fmt.Print("a->") fmt.Scan(&a) fmt.Println(digitcountsum(a)) } } </pre>	

- 4) Описать процедуру Swap(X, Y), меняющую содержимое переменных X и Y (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных A, B, C, D последовательно поменять содержимое следующих пар: A и B, C и D, B и C и вывести новые значения A, B, C, D .

PascalABC.NET	C++
<pre> procedure Swap(var X,Y: Real); begin (X,Y):=(Y,X); end; var A,B,C,D:Real; begin Write('A,B,C,D :'); Readln(A,B,C,D); Swap(A,B); Swap(C,D); Swap(B,C); Writeln(A, ' ',B, ' ',C, ' ',D); end. </pre>	<pre> #include <iostream> using namespace std; void swap(int &a, int &b) { int h = a; a = b; b = h; } int main(){ int a, b, c, d; cin >> a >> b >> c >> d; swap(a, b); swap(c, d); swap(b, c); cout << a << " " << b << " " << c << " " << d << endl; system("pause"); return 0; } </pre>
Go	
<pre> package main import "fmt" func swap(x,y int) (int, int) { return y,x } func main() { var a,b,c,d int fmt.Print("a,b,c,d->") fmt.Scan(&a,&b,&c,&d) fmt.Println(swap(a,b)) fmt.Println(swap(c,d)) fmt.Println(swap(b,c)) } </pre>	
Kotlin	
<pre> import java.util.Scanner fun swap(a: Float, b:Float): Unit { println("b=\$a a=\$b") } fun main(args: Array<String>){ val reader = Scanner(System.`in`) var a=reader.nextFloat() var b=reader.nextFloat() var c=reader.nextFloat() var d=reader.nextFloat() swap(a,b) swap(c,d) swap(b,c) } </pre>	

2. Функции с числовыми параметрами

5) Описать функцию $\text{Sign}(X)$ целого типа, возвращающую для вещественного числа X следующие значения:

-1, если $X < 0$; 0, если $X = 0$; 1, если $X > 0$.

С помощью этой функции найти значение выражения $\text{Sign}(A) + \text{Sign}(B)$ для данных вещественных чисел A и B .

PascalABC.NET	C++
<pre>function Sign(X: Real): Integer; begin if X<0 then Sign:=-1; if X=0 then Sign:=0; if X>0 then Sign:=1; end; var A, B: Real; begin Write('A,B :'); Readln(A, B); Writeln(Sign(A) + Sign(B)); end.</pre>	<pre>#include <iostream> using namespace std; int sign(float x) { if (x<0) return -1; if (x == 0) return 0; if (x>0) return 1; } int main(){ float a, b; cin >> a >> b; int c = sign(a) + sign(b); cout << c<<endl; system("pause"); return 0; }</pre>
Python	C#
<pre>def sign(x): if x<0: return -1 elif x>0: return 1 else: return 0 a=float(input("a")) b=float(input("b")) S=sign(a)+sign(b) print(S)</pre>	<pre>package main import "fmt" func sign(x float64) (int) { if x<0{ return -1 } else if x>0{ return 1 } else{ return 0 } } func main() { var a, b float64 fmt.Print("a,b->") fmt.Scan(&a,&b) var c=sign(a)+sign(b) fmt.Println(c) }</pre>
Kotlin	
<pre>import java.util.Scanner fun sign(a: Float) :Int{ if (a<0)</pre>	

```

return -1
else if (a>0)
    return 1
else
    return 0
}
fun main(args: Array<String>) {
    val reader = Scanner(System.`in`)
    var a = reader.nextFloat()
    var b = reader.nextFloat()
    var c = sign(a) + sign(b);
    println(c)
}

```

VB.NET

```

Function Sign(x As Single) As Integer
    If x > 0 Then
        Return 1
    ElseIf x < 0 Then
        Return -1
    Else
        Return 0
    End If
End Function
Sub Main()
    Dim a, b As Single
    Dim m As Integer
    a = Convert.ToSingle(Console.ReadLine())
    b = Convert.ToSingle(Console.ReadLine())
    m = Sign(a) + Sign(b)
    Console.WriteLine(m)
    Console.ReadKey()
End Sub

```

6) Описать функцию SumRange(A, B) целого типа, находящую сумму всех целых чисел от A до B включительно (A и B — целые). Если $A > B$, то функция возвращает 0. С помощью этой функции найти суммы чисел от A до B и от B до C , если даны числа A, B, C .

PascalABC.NET	C++
<pre> function SumRange(A, B: Integer): Integer; var i, S: Integer; begin S := 0; for i := A to B do begin S := S + i; end; SumRange := S; end; </pre>	<pre> #include <iostream> using namespace std; int sumrange(int a, int b) { int sum = 0; for (; a <= b; ++a) sum += a; return sum; } int main(){ int a, b, c, S, S1; cin >> a >> b>>c; S = sumrange(a, b); </pre>

<pre> var A, B, C: Integer; begin Write('A,B,C :'); Readln(A, B, C); Writeln(SumRange(A, B)); Writeln(SumRange(B, C)); end.</pre>	<pre> S1 = sumrange(b, c); cout <<S<<" "<<S1<<endl; system("pause"); return 0; }</pre>
Go	
<pre> package main import "fmt" func sum(a,b int) (int) { var s=0 for ;a<=b; a++){ s+=a } return s } func main() { var a, b,c int fmt.Print("a,b,c->") fmt.Scan(&a,&b, &c) fmt.Println(sum(a,b)) fmt.Println(sum(b,c)) }</pre>	

7) Описать функцию IsPrime(N) логического типа, возвращающую True, если целый параметр N (> 1) является простым числом, и False в противном случае (число, большее 1, называется *простым*, если оно не имеет положительных делителей, кроме 1 и самого себя). Дан набор из 10 целых чисел, больших 1. С помощью функции IsPrime найти количество простых чисел в данном наборе.

PascalABC.NET	C++
<pre> function IsPrime(N: Integer): Boolean; var i: integer; begin i := 1; IsPrime := True; while(i<=trunc(sqrt(N)))do begin Inc(i); if N mod i = 0 then begin I := N; IsPrime := False; end; end;</pre>	<pre> #include <iostream> #include <cmath> using namespace std; int isprime(int n) { int i, p = 1; for (i=2;i<=(int)sqrt(n);i++) if (n%i == 0) { p = 0; break; } return p; } int main(){ int i, n, res = 0; for (i = 1; i <= 10; ++i) {</pre>

<pre> end; end; var i, N, Res: Integer; begin Res := 0; for i := 1 to 10 do begin Write('N :'); Readln(N); if IsPrime(N) then Inc(Res); end; Writeln(Res); end. </pre>	<pre> cin >> n; res += isprime(n); } cout << res << endl; system("pause"); return 0; } </pre>
Go	
<pre> package main import "fmt" func isprime(a int) (bool) { var y bool=true for i:=2; i< a/2; i++){ if (a%i==0){ y=false break } } return y } func main() { var a,k int=0,0 fmt.Print("a->") for i:=1; i<=10; i++){ fmt.Scan(&a) if (isprime(a)){ k++ } } fmt.Println(k) } </pre>	

3. Дополнительные задания на процедуры и функции

8) Описать функцию $\text{Exp1}(x, \varepsilon)$ вещественного типа (параметры x, ε — вещественные, $\varepsilon > 0$), находящую приближенное значение функции $\exp(x)$:

$$\exp(x) = 1 + x + x^2/(2!) + x^3/(3!) + \dots + x^n/(n!) + \dots$$

($n! = 1 \cdot 2 \cdot \dots \cdot n$). В сумме учитывать все слагаемые, большие ε . С помощью Exp1 найти приближенное значение экспоненты для данного x при шести данных ε .

PascalABC.NET	C++
<pre> function Exp1(x,e:Real):Real; var i: integer; xi, ni: Real; TempRez, Rez: Real; begin xi := 1; i := 0; ni := 1; Rez := 0; TempRez := 1; while (TempRez > e) do begin Rez := Rez + TempRez; inc(i); xi := xi * x; ni := ni * i; TempRez := xi / ni; end; Exp1 := Rez; end; var X, E: real; i: integer; begin Write('X :'); Readln(X); for i := 1 to 6 do begin Write('E :'); Readln(E); Writeln(Exp1(X, E)); end; end. </pre>	<pre> #include <iostream> using namespace std; float exp1(float x, float e) { float xi=1,ni=1,rez=0,h=1; int i = 0; while (temprez>e) { rez += temprez; ++i; xi *= x; ni *= i; temprez = xi / ni; } return rez; } int main(){ float x, e; cin >> x; for (int i = 1; i <= 6; ++i) { cin >> e; cout << exp1(x, e) << endl; } system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math") func exp1(x,e float64) float64 { var s,p,t, i float64=1,1,1,1 for t>=e{ p=p*i; t=math.Pow(x,i)/p s=s+t </pre>	

```

        i++;
    }
    return s
}
func main() {
    var x,e float64
    fmt.Print("x")
    fmt.Scan(&x)
    for i := 0; i < 10; i++){
        fmt.Scan(&e)
        fmt.Println(exp1(x,e))
    }
}

```

9) Описать процедуру TimeToHMS(T, H, M, S), определяющую по времени T (в секундах) содержащееся в нем количество часов H , минут M и секунд S (T — входной, H, M и S — выходные параметры целого типа). Используя эту процедуру, найти количество часов, минут и секунд для пяти данных отрезков времени T_1, T_2, \dots, T_5 .

PascalABC.NET	C++
<pre> procedure TimeToHMS(T: integer; var H,M,S:integer); begin H := T div 3600; T := T - H * 3600; M := T div 60; S := T - M * 60 end; var T, H, M, S: integer; i: byte; begin for i := 1 to 5 do begin write('T', i, ' = '); readln(T); if T >= 0 then begin TimeToHMS(T, H, M, S); writeln(H, ':', M, ':', S); end else writeln(Error!'); end end. </pre>	<pre> #include <iostream> using namespace std; void TimeToHMS(int T, int &H, int &M, int &S) { H = T / 3600; M = (T % 3600) / 60; S = T % 60; } int main() { int T, H, M, S; for (int i = 1; i <= 5; ++i) { cin >> T; TimeToHMS(T, H, M, S); cout<<H<<":"<<M<<":"<<S<<endl; } system("pause"); return 0; } </pre>

Go

```
package main
import "fmt"
func TimeToHMS(T int) (int, int, int) {
    var H = T / 3600
    var M = (T % 3600) / 60
    var S = T % 60
    return H, M, S
}
func main() {
    var t, h, m, s int
    for i := 0; i < 3; i++ {
        fmt.Print("t->")
        fmt.Scan(&t)
        h, m, s = TimeToHMS(t)
        fmt.Printf("%v:%v:%v\n", h, m, s)
    }
}
```

Задачи для самостоятельной работы

Процедуры с числовыми параметрами

- 1) Описать процедуру $\text{PowerA234}(A, B, C, D)$, вычисляющую вторую, третью и четвертую степень числа A и возвращающую эти степени соответственно в переменных B, C и D (A — входной, B, C, D — выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти вторую, третью и четвертую степень пяти данных чисел.
- 2) Описать процедуру $\text{TrianglePS}(a, P, S)$, вычисляющую по стороне a равностороннего треугольника его периметр $P = 3 \cdot a$ и площадь $S = \frac{a^2 \sqrt{3}}{4}$ (a — входной, P и S — выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти периметры и площади трех равносторонних треугольников с данными сторонами.
- 3) Описать процедуру $\text{RectPS}(x_1, y_1, x_2, y_2, P, S)$, вычисляющую периметр P и площадь S прямоугольника со сторонами, параллельными осям координат, по координатам $(x_1, y_1), (x_2, y_2)$ его противоположных вершин (x_1, y_1, x_2, y_2 — входные, P и S — выходные параметры вещественного типа). С помощью этой процедуры найти периметры и площади трех прямоугольников с данными противоположными вершинами.
- 4) Описать процедуру $\text{InvertDigits}(K)$, меняющую порядок следования цифр целого положительного числа K на обратный (K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел.
- 5) Описать процедуру $\text{AddRightDigit}(D, K)$, добавляющую к целому положительному числу K справа цифру D (D — входной параметр целого типа, лежащий в диапазоне 0–9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры последовательно добавить к данному числу K справа данные цифры D_1 и D_2 , выводя результат каждого добавления.
- 6) Описать процедуру $\text{AddLeftDigit}(D, K)$, добавляющую к целому положительному числу K слева цифру D (D — входной параметр целого типа, лежащий в диапазоне 1–9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры последовательно добавить к данному числу K слева данные цифры D_1 и D_2 , выводя результат каждого добавления.
- 7) Описать процедуру $\text{Minmax}(X, Y)$, записывающую в переменную X минимальное из значений X и Y , а в переменную Y — максимальное из этих значений (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). Используя четыре вызова этой процедуры, найти минимальное и максимальное из данных чисел A, B, C, D .
- 8) Описать процедуру $\text{SortInc3}(A, B, C)$, меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по возрастанию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по возрастанию два данных набора из трех чисел: (A_1, B_1, C_1) и (A_2, B_2, C_2) .
- 9) Описать процедуру $\text{SortDec3}(A, B, C)$, меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по убыванию ($A,$

B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по убыванию два данных набора из трех чисел: (A_1, B_1, C_1) и (A_2, B_2, C_2) .

10) Описать процедуру $\text{ShiftRight3}(A, B, C)$, выполняющую *правый циклический сдвиг*: значение A переходит в B , значение B — в C , значение C — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры выполнить правый циклический сдвиг для двух данных наборов из трех чисел: (A_1, B_1, C_1) и (A_2, B_2, C_2) .

11) Описать процедуру $\text{ShiftLeft3}(A, B, C)$, выполняющую *левый циклический сдвиг*: значение A переходит в C , значение C — в B , значение B — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры выполнить левый циклический сдвиг для двух данных наборов из трех чисел: (A_1, B_1, C_1) и (A_2, B_2, C_2) .

Функции с числовыми параметрами

12) Описать функцию $\text{RootsCount}(A, B, C)$ целого типа, определяющую количество корней квадратного уравнения $A \cdot x^2 + B \cdot x + C = 0$ (A, B, C — вещественные параметры, $A \neq 0$). С ее помощью найти количество корней для каждого из трех квадратных уравнений с данными коэффициентами. Количество корней определять по значению *дискриминанта*: $D = B^2 - 4 \cdot A \cdot C$.

13) Описать функцию $\text{CircleS}(R)$ вещественного типа, находящую площадь круга радиуса R (R — вещественное). С помощью этой функции найти площади трех кругов с данными радиусами. Площадь круга радиуса R вычисляется по формуле $S = \pi \cdot R^2$. В качестве значения π использовать 3.14.

14) Описать функцию $\text{RingS}(R_1, R_2)$ вещественного типа, находящую площадь кольца, заключенного между двумя окружностями с общим центром и радиусами R_1 и R_2 (R_1 и R_2 — вещественные, $R_1 > R_2$). С ее помощью найти площади трех колец, для которых даны внешние и внутренние радиусы. Воспользоваться формулой площади круга радиуса R : $S = \pi \cdot R^2$. В качестве значения π использовать 3.14.

15) Описать функцию $\text{TriangleP}(a, h)$, находящую периметр равнобедренного треугольника по его основанию a и высоте h , проведенной к основанию (a и h — вещественные). С помощью этой функции найти периметры трех треугольников, для которых даны основания и высоты. Для нахождения боковой стороны b треугольника использовать *теорему Пифагора*:

$$b^2 = (a/2)^2 + h^2.$$

16) Описать функцию $\text{Calc}(A, B, Op)$ вещественного типа, выполняющую над ненулевыми вещественными числами A и B одну из арифметических операций и возвращающую ее результат. Вид операции определяется целым параметром Op : 1 — вычитание, 2 — умножение, 3 — деление, остальные значения — сложение. С помощью Calc выполнить для данных A и B операции, определяемые данными целыми N_1, N_2, N_3 .

17) Описать функцию $\text{Quarter}(x, y)$ целого типа, определяющую номер координатной четверти, в которой находится точка с ненулевыми вещественными координатами (x, y) . С помощью этой функции найти номера координатных четвертей для трех точек с данными ненулевыми координатами.

- 18) Описать функцию $\text{Even}(K)$ логического типа, возвращающую True, если целый параметр K является четным, и False в противном случае. С ее помощью найти количество четных чисел в наборе из 10 целых чисел.
- 19) Описать функцию $\text{IsSquare}(K)$ логического типа, возвращающую True, если целый параметр $K (> 0)$ является квадратом некоторого целого числа, и False в противном случае. С ее помощью найти количество квадратов в наборе из 10 целых положительных чисел.
- 20) Описать функцию $\text{IsPower5}(K)$ логического типа, возвращающую True, если целый параметр $K (> 0)$ является степенью числа 5, и False в противном случае. С ее помощью найти количество степеней числа 5 в наборе из 10 целых положительных чисел.
- 21) Описать функцию $\text{IsPowerN}(K, N)$ логического типа, возвращающую True, если целый параметр $K (> 0)$ является степенью числа $N (> 1)$, и False в противном случае. Дано число $N (> 1)$ и набор из 10 целых положительных чисел. С помощью функции IsPowerN найти количество степеней числа N в данном наборе.
- 22) Описать функцию $\text{DigitCount}(K)$ целого типа, находящую количество цифр целого положительного числа K . Используя эту функцию, найти количество цифр для каждого из пяти данных целых положительных чисел.
- 23) Описать функцию $\text{DigitN}(K, N)$ целого типа, возвращающую N -ю цифру целого положительного числа K (цифры в числе нумеруются справа налево). Если количество цифр в числе K меньше N , то функция возвращает -1 . Для каждого из пяти данных целых положительных чисел K_1, K_2, \dots, K_5 вызвать функцию DigitN с параметром N , изменяющимся от 1 до 5.
- 24) Описать функцию $\text{IsPalindrom}(K)$, возвращающую True, если целый параметр $K (> 0)$ является *палиндромом* (то есть его запись читается одинаково слева направо и справа налево), и False в противном случае. С ее помощью найти количество палиндромов в наборе из 10 целых положительных чисел. При описании функции можно использовать функции DigitCount и DigitN .
- 25) Описать функцию $\text{DegToRad}(D)$ вещественного типа, находящую величину угла в радианах, если дана его величина D в градусах (D — вещественное число, $0 < D < 360$). Воспользоваться следующим соотношением: $180^\circ = \pi$ радианов. В качестве значения π использовать 3.14. С помощью функции DegToRad перевести из градусов в радианы пять данных углов.
- 26) Описать функцию $\text{RadToDeg}(R)$ вещественного типа, находящую величину угла в градусах, если дана его величина R в радианах (R — вещественное число, $0 < R < 2 \cdot \pi$). Воспользоваться следующим соотношением: $180^\circ = \pi$ радианов. В качестве значения π использовать 3.14. С помощью функции RadToDeg перевести из радианов в градусы пять данных углов.
- 27) Описать функцию $\text{Fact}(N)$ вещественного типа, вычисляющую значение *факториала* $N! = 1 \cdot 2 \cdot \dots \cdot N$ ($N > 0$ — параметр целого типа; вещественное возвращаемое значение используется для того, чтобы избежать целочисленного переполнения при больших значениях N). С помощью этой функции найти факториалы пяти данных целых чисел.
- 28) Описать функцию $\text{Fact2}(N)$ вещественного типа, вычисляющую *двойной факториал*:
- $$N!! = 1 \cdot 3 \cdot 5 \cdot \dots \cdot N, \quad \text{если } N \text{ — нечетное;}$$
- $$N!! = 2 \cdot 4 \cdot 6 \cdot \dots \cdot N, \quad \text{если } N \text{ — четное}$$

($N > 0$ — параметр целого типа; вещественное возвращаемое значение используется для того, чтобы избежать целочисленного переполнения при больших значениях N). С помощью этой функции найти двойные факториалы пяти данных целых чисел.

- 29) Описать функцию $\text{Fib}(N)$ целого типа, вычисляющую N -й элемент последовательности чисел Фибоначчи F_K , которая описывается следующими формулами:

$$F_1 = 1, \quad F_2 = 1, \quad F_K = F_{K-2} + F_{K-1}, \quad K = 3, 4, \dots$$

Используя функцию Fib , найти пять чисел Фибоначчи с данными номерами N_1, N_2, \dots, N_5 .

- 30) Описать функцию $\text{Power1}(A, B)$ вещественного типа, находящую величину A^B по формуле $A^B = \exp(B \cdot \ln(A))$ (параметры A и B — вещественные). В случае нулевого или отрицательного параметра A функция возвращает 0. С помощью этой функции найти степени A^P, B^P, C^P , если даны числа P, A, B, C .

- 31) Описать функцию $\text{Power2}(A, N)$ вещественного типа, находящую величину A^N (A — вещественный, N — целый параметр) по следующим формулам:

$$A^0 = 1;$$

$$A^N = A \cdot A \cdot \dots \cdot A \quad (N \text{ сомножителей}), \quad \text{если } N > 0;$$

$$A^N = 1/(A \cdot A \cdot \dots \cdot A) \quad (|N| \text{ сомножителей}), \quad \text{если } N < 0.$$

С помощью этой функции найти A^K, A^L, A^M , если даны числа A, K, L, M .

- 32) Используя функции Power1 и Power2 , описать функцию $\text{Power3}(A, B)$ вещественного типа с вещественными параметрами, находящую A^B следующим образом: если B имеет нулевую дробную часть, то вызывается функция $\text{Power2}(A, \text{Round}(B))$; в противном случае вызывается функция $\text{Power1}(A, B)$. С помощью этой функции найти A^P, B^P, C^P , если даны числа P, A, B, C .

Дополнительные задания на процедуры и функции

- 33) Описать функцию $\text{Sin1}(x, \varepsilon)$ вещественного типа (параметры x, ε — вещественные, $\varepsilon > 0$), находящую приближенное значение функции $\sin(x)$:

$$\sin(x) = x - x^3/(3!) + x^5/(5!) - \dots + (-1)^n \cdot x^{2 \cdot n + 1} / ((2 \cdot n + 1)!) + \dots$$

В сумме учитывать все слагаемые, модуль которых больше ε . С помощью Sin1 найти приближенное значение синуса для данного x при шести данных ε .

- 34) Описать функцию $\text{Cos1}(x, \varepsilon)$ вещественного типа (параметры x, ε — вещественные, $\varepsilon > 0$), находящую приближенное значение функции $\cos(x)$:

$$\cos(x) = 1 - x^2/(2!) + x^4/(4!) - \dots + (-1)^n \cdot x^{2 \cdot n} / ((2 \cdot n)!) + \dots$$

В сумме учитывать все слагаемые, модуль которых больше ε . С помощью Cos1 найти приближенное значение косинуса для данного x при шести данных ε .

- 35) Описать функцию $\text{Ln1}(x, \varepsilon)$ вещественного типа (параметры x, ε — вещественные, $|x| < 1, \varepsilon > 0$), находящую приближенное значение функции $\ln(1 + x)$:

$$\ln(1 + x) = x - x^2/2 + x^3/3 - \dots + (-1)^n \cdot x^{n+1} / (n+1) + \dots$$

В сумме учитывать все слагаемые, модуль которых больше ε . С помощью Ln1 найти приближенное значение $\ln(1 + x)$ для данного x при шести данных ε .

- 36) Описать функцию $\text{Arctg1}(x, \varepsilon)$ вещественного типа (параметры x, ε — вещественные, $|x| < 1, \varepsilon > 0$), находящую приближенное значение функции $\text{arctg}(x)$:

$$\text{arctg}(x) = x - x^3/3 + x^5/5 - \dots + (-1)^n \cdot x^{2 \cdot n + 1} / (2 \cdot n + 1) + \dots$$

В сумме учитывать все слагаемые, модуль которых больше ε . С помощью `Arctg1` найти приближенное значение $\text{arctg}(x)$ для данного x при шести данных ε .

- 37) Описать функцию `Power4(x, a, ε)` вещественного типа (параметры x, a, ε — вещественные, $|x| < 1; a, \varepsilon > 0$), находящую приближенное значение функции $(1 + x)^a$:

$$(1 + x)^a = 1 + a \cdot x + a \cdot (a-1) \cdot x^2 / (2!) + \dots + a \cdot (a-1) \cdot \dots \cdot (a-n+1) \cdot x^n / (n!) + \dots$$

В сумме учитывать все слагаемые, модуль которых больше ε . С помощью `Power4` найти приближенное значение $(1 + x)^a$ для данных x и a при шести данных ε .

- 38) Описать функцию `NOD2(A, B)` целого типа, находящую *наибольший общий делитель* (НОД) двух целых положительных чисел A и B , используя *алгоритм Евклида*:

$$\text{НОД}(A, B) = \text{НОД}(B, A \bmod B), \text{ если } B \neq 0; \quad \text{НОД}(A, 0) = A.$$

С помощью этой функции найти наибольшие общие делители пар (A, B) , (A, C) , (A, D) , если даны числа A, B, C, D .

- 39) Используя функцию `NOD2`, описать процедуру `Frac1(a, b, p, q)`, преобразующую дробь a/b к несократимому виду p/q (все параметры процедуры — целого типа, a и b — входные, p и q — выходные). Знак результирующей дроби p/q приписывается числителю (то есть $q > 0$). С помощью `Frac1` найти несократимые дроби, равные $a/b + c/d$, $a/b + e/f$, $a/b + g/h$ (числа a, b, c, d, e, f, g, h даны).

- 40) Учитывая, что *наименьшее общее кратное* двух целых положительных чисел A и B равно $A \cdot (B / \text{НОД}(A, B))$, где $\text{НОД}(A, B)$ — наибольший общий делитель A и B , и используя функцию `NOD2`, описать функцию `NOK2(A, B)` целого типа, находящую наименьшее общее кратное чисел A и B . С помощью `NOK2` найти наименьшие общие кратные пар (A, B) , (A, C) , (A, D) , если даны числа A, B, C, D .

- 41) Учитывая соотношение $\text{НОД}(A, B, C) = \text{НОД}(\text{НОД}(A, B), C)$ и используя функцию `NOD2`, описать функцию `NOD3(A, B, C)` целого типа, находящую наибольший общий делитель трех целых положительных чисел A, B, C . С помощью этой функции найти наибольшие общие делители троек (A, B, C) , (A, C, D) и (B, C, D) , если даны числа A, B, C, D .

- 42) Описать процедуру `IncTime(H, M, S, T)`, которая увеличивает на T секунд время, заданное в часах H , минутах M и секундах S (H, M и S — входные и выходные параметры, T — входной параметр; все параметры — целые положительные). Дано время (в часах H , минутах M , секундах S) и целое число T . Используя процедуру `IncTime`, увеличить данное время на T секунд и вывести новые значения H, M, S .

- 43) Описать функцию `IsLeapYear(Y)` логического типа, которая возвращает `True`, если год Y (целое положительное число) является високосным, и `False` в противном случае. Вывести значение функции `IsLeapYear` для пяти данных значений параметра Y . *Високосным* считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400.

- 44) Используя функцию `IsLeapYear`, описать функцию `MonthDays(M, Y)` целого типа, которая возвращает количество дней для M -го месяца года Y ($1 \leq M \leq 12, Y > 0$ — целые числа). Вывести значение функции `MonthDays` для данного года Y и месяцев M_1, M_2, M_3 .

45) Используя функцию MonthDays, описать процедуру PrevDate(D, M, Y), которая по информации о правильной дате, включающей день D , номер месяца M и год Y , определяет предыдущую дату (параметры целого типа D, M, Y являются одновременно входными и выходными). Применить процедуру PrevDate к трем исходным датам и вывести полученные значения предыдущих дат.

46) Используя функцию MonthDays, описать процедуру NextDate(D, M, Y), которая по информации о правильной дате, включающей день D , номер месяца M и год Y , определяет следующую дату (параметры целого типа D, M, Y являются одновременно входными и выходными). Применить процедуру NextDate к трем исходным датам и вывести полученные значения следующих дат.

47) Описать функцию Leng(x_A, y_A, x_B, y_B) вещественного типа, находящую длину отрезка AB на плоскости по координатам его концов:

$$|AB| = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

(x_A, y_A, x_B, y_B — вещественные параметры). С помощью этой функции найти длины отрезков AB, AC, AD , если даны координаты точек A, B, C, D .

48) Используя функцию Leng, описать функцию Perim($x_A, y_A, x_B, y_B, x_C, y_C$) вещественного типа, находящую периметр треугольника ABC по координатам его вершин ($x_A, y_A, x_B, y_B, x_C, y_C$ — вещественные параметры). С помощью этой функции найти периметры треугольников ABC, ABD, ACD , если даны координаты точек A, B, C, D .

49) Используя функции Leng и Perim, описать функцию Area($x_A, y_A, x_B, y_B, x_C, y_C$) вещественного типа, находящую площадь треугольника ABC по формуле

$$S_{ABC} = \sqrt{p \cdot (p - |AB|) \cdot (p - |AC|) \cdot (p - |BC|)},$$

где p — полупериметр. С помощью этой функции найти площади треугольников ABC, ABD, ACD , если даны координаты точек A, B, C, D .

50) Используя функции Leng и Area, описать функцию Dist($x_P, y_P, x_A, y_A, x_B, y_B$) вещественного типа, находящую расстояние $D(P, AB)$ от точки P до прямой AB по формуле

$$D(P, AB) = 2 \cdot S_{PAB} / |AB|,$$

где S_{PAB} — площадь треугольника PAB . С помощью этой функции найти расстояния от точки P до прямых AB, AC, BC , если даны координаты точек P, A, B, C .

51) Используя функцию Dist, описать процедуру Heights($x_A, y_A, x_B, y_B, x_C, y_C, h_A, h_B, h_C$), находящую высоты h_A, h_B, h_C треугольника ABC (выходные параметры), проведенные соответственно из вершин A, B, C (их координаты являются входными параметрами). С помощью этой процедуры найти высоты треугольников ABC, ABD, ACD , если даны координаты точек A, B, C, D .

10. Минимумы и максимумы: группа Minmax

Во всех заданиях данной группы предполагается, что исходный набор содержит ненулевое количество элементов (в частности, число N всегда больше нуля).

Для решения заданий из данной группы, как и для заданий группы Series, следует использовать «однопроходные» алгоритмы, позволяющие получить требуемый результат после *однократного* просмотра набора исходных данных.

- 1) Дано целое число N и набор из N чисел. Найти минимальный и максимальный из элементов данного набора и вывести их в указанном порядке.

PascalABC.NET	C++
<pre> Var R, Min, Max: Real; i, N: Integer; begin Write('N:'); Readln(N); for i := 1 to N do begin Readln(R); if i = 1 then begin Min := R; Max := R; end; if R<Min then Min:= R; if R>Max then Max:= R; end; Writeln(Min, ' ', Max); End. </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i; float a, min, max; cin >> n >> a; min = max = a; for (i = 2; i <= n; ++i) { cin >> a; if (i == 1) { min = a; max = a; } if (a < min) min = a; if (a > max) max = a; } cout<<"min="<<min<<" max="<<max <<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> n=int(input("->")) a=float(input("->")) mx=a mn=a for i in range(1,n): a=float(input("a")) if a>mx: mx=a if a<mn: mn=a print(mx,mn) </pre>	<pre> static void Main(string[] args) { double a, mx, mn; int n; n=Convert.ToInt32(Console.ReadLine()); a=Convert.ToDouble(Console.ReadLine()); mx=mn=a; for (int i=2; i<=n; i++){ a=Convert.ToDouble(Console.ReadLine()); if (a>mx) mx=a; if (a<mn) mn=a; } Console.WriteLine("max={0} min={1}", mx, mn); Console.ReadKey(); } </pre>

Java	Go
<pre>public static void main(String[] args){ Scanner sc=new Scanner(System.in); double a, max, min; a=sc.nextDouble(); int n=sc.nextInt(); min = max = a; for (int i = 2; i <= n; ++i) { a=sc.nextDouble(); if (a < min) min = a; if (a > max) max = a; } System.out.println("min="+min+" max="+max); }</pre>	<pre>package main import "fmt" func main() { n:=0; fmt.Scan(&n) var a=0.0 fmt.Scan(&a) mx:=a mn:=a for i:=2; i<=n; i++){ fmt.Scan(&a) if (a>mx){ mx=a } if (a<mn){ mn=a } } fmt.Println(mx, mn) }</pre>
Kotlin	
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var n = reader.nextInt() var a = reader.nextDouble() var max = a var min = a for (i in 2..n) { val a = reader.nextDouble() if (a > max) max = a; if (a < min) min = a; } println("min=\$min max=\$max") }</pre>	
VB.NET	
<pre>Sub Main() Dim i, n As Integer Dim a, mn, mx As Single n = Convert.ToInt32(Console.ReadLine()) a = Convert.ToSingle(Console.ReadLine()) mx = a mn = a For i = 2 To n a = Convert.ToSingle(Console.ReadLine()) If mx < a Then mx = a If mn > a Then</pre>	

```

                mn = a
            End If
        End If
    Next
    Console.WriteLine("min={0} max={1}", mn, mx)
    Console.ReadKey()
End Sub

```

2) Дано целое число N и набор из N прямоугольников, заданных своими сторонами — парами чисел (a, b) . Найти максимальный периметр прямоугольника из данного набора.

PascalABC.NET	C++
<pre> Var P, a, b, Min: Real; i, N: Integer; begin Write('N:'); Readln(N); for i := 1 to N do begin Write('a b:'); Readln(a, b); P := 2 * (a + b); if i = 1 then Min := P; if P < Min then Min := P; end; Writeln(Min); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int n, i; float a, b, p, min, max; cin >> n; for (i = 1; i <= n; ++i) { cin >> a >> b; p = 2 * (a + b); if (i == 1) { min = p; max = p; } if (p < min) min = p; if (p > max) max = p; } cout << "min=" << min << " max=" << max << endl; system("pause"); return 0; } </pre>

3) Дано целое число N и набор из N целых чисел. Найти номера первого минимального и последнего максимального элемента из данного набора и вывести их в указанном порядке.

PascalABC.NET	C++
<pre> Var R, Min, Max: Real; i, N, MinNum, MaxNum: Integer; begin Write('N:'); Readln(N); for i := 1 to N do begin Write(i, ' число:'); Readln(R); if i = 1 then begin Min := R; </pre>	<pre> #include <iostream> using namespace std; int main() { int i, n, minnum=1, maxnum=1, a, min, max; cin >> n; cin >> a; max = min = a; for (i = 2; i <= n; ++i) { cin >> a; if (a < min) { min = a; minnum = i; } </pre>

<pre> Max := R; MaxNum := i; MinNum := i; end; if R < Min then begin Min := R; MinNum := i; end; if R >= Max then begin Max := R; MaxNum := i; end; end; Write(MinNum, ' ',MaxNum); end. </pre>	<pre> if (a >= max) { max = a; maxnum = i; } } cout << "min=" << min << " minnum=" << minnum << endl; cout << "max=" << max << " maxnum=" << maxnum << endl; system("pause"); return 0; } </pre>
---	---

- 4) Дано целое число N и набор из N целых чисел. Найти количество элементов, содержащихся между первым и последним максимальным элементом. Если в наборе имеется единственный максимальный элемент, то вывести 0.

PascalABC.NET	C++
<pre> Var N, Ni, Max, MaxNum, i, RezNum: Integer; begin Write('N:'); Readln(N); for i := 1 to N do begin Write(i, ': '); Readln(Ni); if (Ni>Max) or (i=1) then begin Max := Ni; MaxNum := i; RezNum := 0; end else if Ni = Max then begin RezNum:=i-1-MaxNum; end; end; Writeln(RezNum); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int i, n, maxnum1=1, maxnum2=1, a, max1, max2; cin >> n; cin >> a; max1 = max2 = a; for (i = 2; i <= n; ++i) { cin >> a; if (a>max1) { max1 = a; maxnum1 = i; } if (a >= max2) { max2 = a; maxnum2= i; } } cout<<(maxnum2-maxnum1)-1<<endl; system("pause"); return 0; } </pre>

- 5) Дано целое число $N (> 2)$ и набор из N чисел. Найти два наименьших элемента из данного набора и вывести эти элементы в порядке возрастания их значений.

PascalABC.NET	C++
<pre> Var Min1, Min2, R, Sum: Real; N, i: Integer; begin Sum := 0; Write('N:'); Readln(N); for i := 1 to N do begin Write(i, ' число:'); Readln(R); if i = 1 then begin Min1 := R; Min2 := R; end; if R < Min2 then if R < Min1 then begin Min2 := Min1; Min1 := R; end else if R <> Min1 then Min2 := R; if Min1 = Min2 then Min2 := R; end; writeln(Min1, ' ', Min2); end. </pre>	<pre> #include <iostream> using namespace std; int main() { float min1, min2, a; int n, i; cin >> n >> a; min1 = min2 = a; for (i = 1; i <= n; ++i) { cin >> a; if ((a < min2) && (a != min1)) { if (a < min1) { min2 = min1; min1 = a; } else min2 = a; } if (min1 == min2) min2 = a; } cout << "min1=" << min1 << " min2=" << min2 << endl; system("pause"); return 0; } </pre>

6) Дано целое число N и набор из N целых чисел, содержащий только нули и единицы. Найти номер элемента, с которого начинается самая длинная последовательность одинаковых чисел, и количество элементов в этой последовательности. Если таких последовательностей несколько, то вывести номер первой из них.

PascalABC.NET
<pre> var N, i, Num, Numi, MaxNum, Maxi: Integer; B, B2: byte; begin B := 5; B2 := 5; Num := 0; Numi := 0; MaxNum := 0; Write('N:'); </pre>

```

Readln(N);
for i := 1 to N do
begin
  B2 := B;
  Write(i, ' число:');
  Readln(B);

  if B2 = B then inc(Num)
  else
  begin
    if Num > MaxNum then
    begin
      MaxNum := Num;
      Maxi := Numi;
    end;
    Numi := i;
    Num := 1;
  end;
end;
if Num > MaxNum then
begin
  MaxNum := Num;
  Maxi := Numi;
end;
Writeln(Maxi, ' ', MaxNum);
end.

```

C++

```

#include <iostream>
using namespace std;
int main()
{
  int b=2,b2=2,num=0,numi=0, maxnum = 0, maxi, n, i;
  cin >> n >> b;
  for (i = 1; i <= n; ++i) {
    b2 = b;
    cin >> b;
    if (b2 == b) ++num;
    else
    {
      if (num > maxnum) {
        maxnum = num;
        maxi = numi;
      }
      numi = i;
      num = 1;
    }
  }
  if (num > maxnum)
  {
    maxnum = num;
  }
}

```



```

        maxi = numi;
    }
    cout << "maxi=" << maxi << " maxnum=" << maxnum << endl;
    system("pause");
    return 0;
}

```

7) Дано целое число N и набор из N целых чисел. Найти максимальное количество подряд идущих минимальных элементов из данного набора.

PascalABC.NET	C++
<pre> Var N,i,Ni,Nmin,Num,MaxNum:Integer; begin Num := 0; MaxNum := 0; Write('N:'); Readln(N); for i := 1 to N do begin Write(i, ' число:'); Readln(Ni); if (i=1) or (Ni<Nmin) then begin Nmin := Ni; Num := 0; MaxNum := 0; end; if Ni = Nmin then inc(Num) else begin if Num > MaxNum then MaxNum := Num; Num := 0; end; end; if (Num>MaxNum) and (ni=Nmin) then MaxNum := Num; Writeln(MaxNum); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int n,i,a,nmin,num,maxnum; cin >> n >> a; nmin = a; num = 0; maxnum = 0; for (i = 2; i <= n; ++i) { cin >> a; if (a < nmin) { nmin = a; num = 0; maxnum = 0; } if (a == nmin) ++num; else { if (num > maxnum) maxnum = num; num = 0; } } if ((num>maxnum) && (a==nmin)) maxnum = num; cout << maxnum << endl; system("pause"); return 0; } </pre>

Задачи для самостоятельной работы

- 1) Дано целое число N и набор из N прямоугольников, заданных своими сторонами — парами чисел (a, b) . Найти минимальную площадь прямоугольника из данного набора.
- 2) Дано целое число N и набор из N чисел. Найти номер минимального элемента из данного набора.
- 3) Дано целое число N и набор из N пар чисел (m, v) — данные о массе m и объеме v деталей, изготовленных из различных материалов. Вывести номер детали, изготовленной из материала максимальной плотности, а также величину этой максимальной плотности. Плотность P вычисляется по формуле $P = m/v$.
- 4) Дано целое число N и набор из N целых чисел. Найти номера первого максимального и последнего минимального элемента из данного набора и вывести их в указанном порядке.
- 5) Дано целое число N и набор из N целых чисел. Найти номера первого и последнего минимального элемента из данного набора и вывести их в указанном порядке.
- 6) Дано целое число N и набор из N целых чисел. Найти номера первого и последнего максимального элемента из данного набора и вывести их в указанном порядке.
- 7) Дано целое число N и набор из N целых чисел. Найти номер первого *экстремального* (то есть минимального или максимального) элемента из данного набора.
- 8) Дано целое число N и набор из N целых чисел. Найти номер последнего *экстремального* (то есть минимального или максимального) элемента из данного набора.
- 9) Дано целое число N и набор из N чисел. Найти минимальное положительное число из данного набора. Если положительные числа в наборе отсутствуют, то вывести 0.
- 10) Дано целое число N и набор из N целых чисел. Найти номер первого максимального нечетного числа из данного набора. Если нечетные числа в наборе отсутствуют, то вывести 0.
- 11) Дано число $B (> 0)$ и набор из десяти чисел. Вывести минимальный из тех элементов набора, которые больше B , а также его номер. Если чисел, больших B , в наборе нет, то дважды вывести 0.
- 12) Даны числа $B, C (0 < B < C)$ и набор из десяти чисел. Вывести максимальный из элементов набора, содержащихся в интервале (B, C) , и его номер. Если требуемые числа в наборе отсутствуют, то дважды вывести 0.
- 13) Дано целое число N и набор из N целых чисел. Найти количество элементов, расположенных перед первым минимальным элементом.
- 14) Дано целое число N и набор из N целых чисел. Найти количество элементов, расположенных после последнего максимального элемента.
- 15) Дано целое число N и набор из N целых чисел. Найти количество минимальных элементов из данного набора.
- 16) Дано целое число N и набор из N целых чисел. Найти общее количество *экстремальных* (то есть минимальных и максимальных) элементов из данного набора.

- 17) Дано целое число $N (> 2)$ и набор из N чисел — значений некоторой величины, полученных в N опытах. Найти среднее значение этой величины. При вычислении среднего значения не учитывать минимальное и максимальное из имеющихся в наборе значений.
- 18) Дано целое число $N (> 3)$ и набор из N чисел. Найти три наибольших элемента из данного набора и вывести эти элементы в порядке убывания их значений.
- 19) Дано целое число $N (> 1)$ и набор из N чисел. Найти максимальную сумму двух соседних чисел из данного набора.
- 20) Дано целое число $N (> 1)$ и набор из N чисел. Найти номера двух соседних чисел из данного набора, произведение которых является минимальным, и вывести вначале меньший, а затем больший номер.
- 21) Дано целое число N и набор из N целых чисел. Найти максимальное количество четных чисел в наборе, идущих подряд. Если четные числа в наборе отсутствуют, то вывести 0.
- 22) Дано целое число N и набор из N целых чисел, содержащий только нули и единицы. Найти номер элемента, с которого начинается самая длинная последовательность единиц, и количество элементов в этой последовательности. Если таких последовательностей несколько, то вывести номер последней из них. Если единицы в исходном наборе отсутствуют, то дважды вывести 0.
- 23) Дано целое число N и набор из N целых чисел. Найти минимальное количество подряд идущих максимальных элементов из данного набора.

11. Одномерные массивы: группа Array

Условие вида «дан массив размера N » означает, что вначале дается *фактический размер* массива (целое число N), а затем приводятся все его элементы. Если в задании явно не указывается, какие значения может принимать размер исходного массива, то предполагается, что размер может изменяться в пределах от 2 до 10. Порядковый номер начального элемента массива считается равным 1.

Если в задании, связанном с созданием (преобразованием) массива, не описан результирующий набор данных, то предполагается, что этим набором является созданный (преобразованный) массив, и необходимо вывести все его элементы в порядке возрастания их индексов.

1. Формирование массива и вывод его элементов

В заданиях на формирование массива предполагается, что размер результирующего массива не превосходит 10.

- 1) Дано целое число $N (> 0)$. Сформировать и вывести целочисленный массив размера N , содержащий N первых положительных нечетных чисел: 1, 3, 5,

PascalABC.NET	C++
<pre> var a: array of integer; N, i: Integer; begin Write('N:'); Readln(N); a:=new integer[N+1]; for i := 1 to N do a[i] := i * 2 - 1; print(a); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int *a,i,n; cin >> n; a = new int[n]; for (i = 0; i<n; ++i) a[i] = 2*i+1; for (i = 0; i<n; ++i) cout<<a[i]<<" "; system("pause"); return 0; } </pre>
Python	C#
<pre> a=[] n=int(input("")) for i in range (1,n): a.append(2*i-1) print(a) </pre>	<pre> static void Main(string[] args){ int n, I; n=Convert.ToInt32(Console.ReadLine()); int []x=new Int32[n]; for (i=0; i<n; i++){ x[i]=2*i+1; Console.Write(x[i]+" "); } Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args){ Scanner sc=new Scanner(System.in); int n, i; n=sc.nextInt(); int [] x=new int[n]; for (i=0; i<n; i++){ </pre>	<pre> package main import "fmt" func main() { var n int fmt.Print("n->") fmt.Scan(&n) </pre>

<pre> x[i]=2*i+1; System.out.print(x[i]+" "); } } </pre>	<pre> var a [] int a=make([]int, n) for i := 0; i < n; i++){ a[i]=2*i+1 } fmt.Println(a) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader=Scanner(System.`in`) var n=reader.nextInt() var x = Array<Int>(n){0} for (i in 0..n-1){ x[i]=2*i-1 } for(i in 0..n-1) { println(x[i]) } } </pre>	<pre> import java.util.Scanner fun main(args: Array<String>) { val reader=Scanner(System.`in`) var n=reader.nextInt() val x = Array(n,{i->2*i+1}) for(i in 0..n-1) { println(x[i]) } } </pre>
VB.NET	
<pre> Sub Main() Dim a() As Integer Dim i, n As Integer n = Convert.ToInt32(Console.ReadLine()) a = New Integer(n) {} For i = 0 To n - 1 a(i) = 2 * i + 1 Console.Write("{0} ", a(i)) Next Console.ReadKey() End Sub </pre>	

2) Дано целое число $N (> 1)$, а также первый член A и разность D арифметической прогрессии. Сформировать и вывести массив размера N , содержащий N первых членов данной прогрессии:

$$A, A + D, A + 2 \cdot D, A + 3 \cdot D, \dots$$

PascalABC.NET	C++
<pre> Var a: array of integer; D, N, i: Integer; begin Write('N:'); Read(N); a:= new integer[N + 1]; Write('A1,D:'); Readln(a[1], D); for i := 2 to N do a[i]:=a[1]+(i-1)*D; for i := 1 to N do writeln(i,':',a[i]); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int *x,i,n, a,d; cin >> n>>a>>d; x= new int[n]; for (i = 0; i<n; ++i) x[i] = a+i*d; for (i = 0; i<n; ++i) cout<<x[i]<<" "; system("pause"); return 0; } </pre>

Python	Go
<pre>N = int(input("N:")) a=[] a.append(int(input("A1:"))) D = int(input("D:")) for i in range(1,N-1): a.append(a[0]+(i)*D) for i in range(N-1): print(a[i], end=' ')</pre>	<pre>package main import "fmt" func main() { var n,a,d int fmt.Print("n,a,d->") fmt.Scan(&n,&a,&d) var x [] int x=make([]int, n) for i := 0; i < n; i++){ x[i]=a+i*d } fmt.Println(x) }</pre>

3) Дан массив размера N . Вывести его элементы в обратном порядке.

PascalABC.NET	C++
<pre>Var a: array of integer; N, i: Integer; begin Write('N:'); Readln(N); a := new integer[N + 1]; for i := 1 to N do begin write(i, ' '); readln(a[i]); end; for i := N downto 1 do writeln(i, ' : ', a[i]); end.</pre>	<pre>#include <iostream> using namespace std; int main(){ int *x,i,n; cin >> n; x= new int[n]; for (i = 0; i < n; ++i) { x[i] = rand() % 50; cout << x[i] << " "; } cout << endl; for (i = n-1; i>=0; i--) cout<<x[i]<<" "; system("pause"); return 0; }</pre>
Python	C#
<pre>n=int(input("n->")) x=[] for i in range(0,n): x.append(int(input("x"))); for i in range(n-1,-1,-1): print(x[i])</pre>	<pre>Вариант 1. static void Main(string[] args) { int n=Convert.ToInt32(Console.ReadLine()); int []x=new Int32[n]; Random rnd=new Random(); for (int i=0; i<n; i++){ x[i] = rnd.Next(30); Console.Write(x[i]+" "); } Console.WriteLine(); for (int i= n - 1; i>= 0; i--) Console.Write(x[i]+" "); Console.ReadKey(); }</pre>

	<pre> Вариант 2. static void Main(string[] args) { int n=Convert.ToInt32(Console.ReadLine()); int []x=new Int32[n]; for (int i=0; i<n; i++) x[i] = Convert.ToInt32(Console.ReadLine()); Console.WriteLine(); for (int i= n - 1; i>= 0; i--) Console.Write(x[i]+" "); Console.ReadKey(); } </pre>
Java	Go
<pre> Вариант 1. public static void main(String[] args){ Scanner sc=new Scanner(System.in); int n, i; n=sc.nextInt(); int [] x=new int[n]; for (i=0; i<n; i++){ x[i]=sc.nextInt(); } for (i=0; i<n; i++) System.out.print(x[i]+" "); System.out.println(); for (i=n-1; i>=0; i--) System.out.print(x[i]+" "); } Вариант 2. public static void main(String[] args){ Scanner sc=new Scanner(System.in); int n, i; n=sc.nextInt(); int [] x=new int[n]; Random rnd=new Random(); for (i=0; i<n; i++){ x[i]=rnd.nextInt(50); System.out.print(x[i]+" "); } System.out.println(); for (i=n-1; i>=0; i--) System.out.print(x[i]+" "); } </pre>	<pre> package main import "fmt" func main() { var n int fmt.Print("n->") fmt.Scan(&n) var x=make([]int, n) for i := 0; i < n; i++ { fmt.Scan(&x[i]) } fmt.Print("PrintArray \n") for i := n-1; i >=0; i--{ fmt.Println(x[i]) } } </pre>

4) Дан массив A размера N и целое число K ($1 \leq K \leq N$). Вывести элементы массива с порядковыми номерами, кратными K : $A_K, A_{2 \cdot K}, A_{3 \cdot K}, \dots$ Условный оператор не использовать.

PascalABC.NET	C++
<pre> Var a: array of integer; N, i, K: Integer; </pre>	<pre> #include <iostream> using namespace std; int main(){ int *x,i,n,k; </pre>

<pre> begin (N, K) := readInteger2; a := new integer[N + 1]; for i := 1 to N do begin write(i, ' '); readln(a[i]); end; i := K; while (i <= N) do begin writeln(i, ':', a[i]); i := i + K; end; end. </pre>	<pre> cin >> n>>k; x= new int[n]; for (i = 0; i < n; ++i) { x[i] = rand() % 50; cout << x[i] << " "; } cout << endl; for (i = k - 1; i<n; i += k) cout<<x[i]<<" "; system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math/rand") func main() { var n,k int fmt.Print("n,k->") fmt.Scan(&n,&k) var x=make([]int, n) for i := 0; i < n; i++ { x[i]=rand.Intn(50); } fmt.Println(x) for i:=k-1; i<n; i+=k{ fmt.Println(x[i]) } } </pre>	

5) Дан массив A размера N. Вывести его элементы в следующем порядке:

$A_1, A_2, A_N, A_{N-1}, A_3, A_4, A_{N-2}, A_{N-3} \dots$

PascalABC.NET
<pre> Var a: array of integer; N, i, P: word; begin readln(N); a := new integer[N + 1]; for i := 1 to N do read(a[i]); writeln(' Результат: '); P := N div 4; for i := 1 to P do write(a[2*i-1], ', ', a[2*i], ', ', a[N+2-2*i], ' ', a[N+1-2*i], ' '); </pre>


```

if N mod 4 > 0 then
  write(a[2 * P + 1], ' ');
if N mod 4 > 1 then
  write(a[2 * P + 2], ' ');
if N mod 4 > 2 then
  write(a[N - 2 * P])
end.

```

C++

```

#include <iostream>
using namespace std;
int main(){
  int *x,i,j,n,p;
  cin >> n;
  x= new int[n];
  for (i = 0; i < n; ++i) {
    x[i] = rand() % 50;
    cout << x[i] << " ";
  }
  cout << endl;
  p = n / 4;
  for (i = 0; i<p; i++)
  cout<<x[2*i]<<" "<<x[2*i+1]<<" "<<x[n-1-2*i]<<" "<<x[n-2-2*i]<<" ";
  if (n % 4 == 1)
    cout << x[(n / 2)] << " ";
  else
    for (j = 0; j < (n % 4); j++)
      cout << x[(n / 2) - 1 + j] << " ";
  system("pause");
  return 0;
}

```

Go

```

package main
import (
  "fmt"
  "math/rand"
)
func main() {
  var n,p int
  fmt.Print("n->")
  fmt.Scan(&n)
  p=n / 4
  var x=make([]int, n)
  for i := 0; i < n; i++ {
    x[i]=rand.Intn(50);
  }
  fmt.Println(x)
  for i:=0; i<p; i++){
    fmt.Printf("%v %v %v %v",x[2*i],x[2*i+1],x[n-1-2*i],
      x[n-2-2*i])
  }
}

```

```

    }
    if (n%4==1){
        fmt.Printf("%+v ", x[(n/2)])
    }else{
        for j:=0; j<(n%4); j++){
            fmt.Printf("%+v ", x[(n/2)-1+j])
        }
    }
}
}

```

2. Анализ элементов массива

Для выполнения некоторых заданий из данного пункта не требуется одновременно хранить в памяти все исходные данные, поэтому использовать при их выполнении массивы, строго говоря, *не нужно*. Однако применение массивов позволяет сделать алгоритмы решения этих заданий более простыми и наглядными. Задания из данного пункта можно дополнить заданиями из групп Series и Minmax, рассматривая их как задания на обработку массивов. С другой стороны, для тех заданий данного пункта, которые можно выполнить, не используя массивы, полезно реализовать и такие алгоритмы решения.

- 6) Дан массив A ненулевых целых чисел размера 10. Вывести значение первого из тех его элементов A_k , которые удовлетворяют неравенству $A_k < A_{10}$. Если таких элементов нет, то вывести 0.

PascalABC.NET	C++
<pre> Var a: array[1..10] of integer; k: Integer; begin for k := 1 to 10 do begin write(k, ' '); readln(a[k]); end; k := 1; while(A[k]>=A[10])and(k<10)do inc(k); if k = 10 then writeln('0') else writeln(a[k]); end. </pre>	<pre> #include <iostream> using namespace std; int main(){ int x[11],i; for (i = 1; i <=10; ++i) { x[i] = rand() % 50; cout << x[i] << " "; } cout << endl; i = 1; while ((x[i]>=x[10])&&(i<10)) i++; if(i == 10) cout << "0"; else cout << x[i] << endl; system("pause"); return 0; } </pre>

- 7) Дан массив A размера N . Найти минимальный элемент из его элементов с четными номерами: A_2, A_4, A_6, \dots

PascalABC.NET	C++
<pre> Var a: array of integer; N, i, Min: Integer; begin Write('N: '); </pre>	<pre> #include <iostream> using namespace std; int main(){ int *x,i, n; cin >> n; </pre>

<pre> Readln(N); a := new integer[N + 1]; for i := 1 to N do begin write(i, ' '); readln(a[i]); end; i := 2; Min := a[2]; while (i <= N) do begin if a[i] < Min then Min := a[i]; i := i + 2; end; Writeln(Min); end.</pre>	<pre> x = new int[n]; for (i = 0; i < n; ++i) { x[i] = rand() % 50; cout << x[i] << " "; } cout << endl; int min = x[0]; for (i = 1; i < n; i += 2) { if (x[i] < min) min = x[i]; } cout << min << endl; system("pause"); return 0; }</pre>
---	--

8) Дан массив размера N . Найти номер его первого локального минимума (локальный минимум — это элемент, который меньше любого из своих соседей).

PascalABC.NET	C++
<pre> Var a:array of integer; N,i:Integer; begin Write('N: '); Readln(N); a:=new integer[N+1]; for i:=1 to N do begin write(i, ' '); readln(a[i]); end; i:=2; if a[1]<a[2] then writeln('1') else begin while (i<=N-1) and (not((a[i-1]>a[i]) and (a[i]<a[i+1]))) do inc(i); Writeln(i); end; end.</pre>	<pre> #include <iostream> using namespace std; int main(){ int *x,i, n; cin >> n; x = new int[n]; for (i = 0; i < n; ++i) { x[i] = rand() % 50; cout << x[i] << " "; } cout << endl; if (x[0] < x[1]) cout << "1" << endl; else { i = 1; while ((i<n-1) && !((x[i-1]> x[i]) && (x[i] < x[i + 1]))) ++i; cout << i + 1 << endl; } system("pause"); return 0; }</pre>

9) Дан массив размера N . Найти количество участков, на которых его элементы монотонно возрастают.

PascalABC.NET
<pre> Var a: array of integer; N, i, Num: Integer; begin</pre>

```

a[1] := 0;
Write('N: ');
Readln(N);
a:=new integer[N + 1];
for i := 1 to N do
begin
  readln(a[i]);
end;
Num := 0;
for i := 3 to N do
  if (a[i - 2] < a[i - 1]) and not (a[i - 1] < a[i]) then
    inc(Num);
if a[N - 1] < a[N] then
  inc(Num);
Writeln(Num);
End.

```

C++

```

#include <iostream>
using namespace std;
int main(){
  int *x,i, n, num=0;
  cin >> n;
  x = new int[n];
  for (i = 0; i <n; ++i)
  {
    x[i] = rand() % 50;
    cout << x[i] << " ";
  }
  cout << endl;
  for (i = 2; i<n; ++i)
    if ((x[i - 2]<x[i - 1]) && !(x[i - 1]<x[i]))
      ++num;
  if (x[n - 2]<x[n - 1])
    ++num;
  cout << num << endl;
  system("pause");
  return 0;
}

```

- 9) Дан целочисленный массив размера N , все элементы которого упорядочены (по возрастанию или по убыванию). Найти количество различных элементов в данном массиве.

PascalABC.NET

```

Var
  i, N, Q: integer;
  A: array of integer;
begin
  write('N= ');
  readln(N);
  a:=new integer[N + 1];
  Q := 1;

```

```

for i := 1 to N do
begin
    read(a[i]);
    if (i > 1) and (a[i] <> a[i - 1]) then inc(Q)
end;
writeln;
writeln('Различных чисел: ', Q)
end.

```

C++

```

#include <iostream>
using namespace std;
int main(){
    int *x, i, n, k = 0;
    bool f;
    cin >> n;
    x = new int[n];
    for (i = 0; i <n; ++i) {
        x[i] = rand() % 5;
        cout << x[i] << " ";
    }
    cout << endl;
    for ( i = 0; i < n; i++){
        int j = 0;
        while (j < i && x[j] != x[i])
            ++j;
        k += j == i;
    }
    cout << k << endl;
    delete x;
    system("pause");
    return 0;
}

```

- 10) Дано число R и массив размера N . Найти два различных элемента массива, сумма которых наиболее близка к числу R , и вывести эти элементы в порядке возрастания их индексов.

PascalABC.NET

```

Var
    A: array of real;
    R, MinDiffSum, c: real;
    i, j, N, i1, i2: byte;
begin
    write('N = ');
    readln(N);
    a := new real[N + 1];
    write('R = ');
    readln(R);
    for i := 1 to N do
        begin
            a[i] := random(10)+1;
            write(' ', a[i]:0:2);
            if i = 2 then
                begin
                    MinDiffSum := abs(a[2] + a[1] - R);
                    i1 := 1;

```

```

    i2 := 2;
end
else
  for j:=1 to i-1 do
  begin
    c := abs(a[j] + a[i] - R);
    if c < MinDiffSum then
    begin
      MinDiffSum := c;
      i1 := j;
      i2 := I;
    end
  end
end;
writeln;
writeln('a[1]:= ',a[i1]:0:2, ' ', 'a[2]:= ',a[i2]:0:2);
end.

```

C++

```

#include <iostream>
using namespace std;
int main(){
  int *x, i,j, r,n, k = 0, k2=1;
  bool f;
  cin >> n>>r;
  x = new int[n];
  for (i = 0; i <n; ++i) {
    x[i] = rand() % 5;
    cout << x[i] << " ";
  }
  cout << endl;
  for (i = 0; i<n - 1; ++i) {
    for (j= i + 1; j<n; ++j) {
      if (abs(x[i] + x[j] - r)<abs(x[k] + x[k2] - r)) {
        k = i;
        k2 = j;
      }
    }
  }
  cout<<k + 1<<" "<<k2 + 1;
  delete x;
  system("pause");
  return 0;
}

```

11) Дан целочисленный массив размера N . Если он является *перестановкой*, то есть содержит все числа от 1 до N , то вывести 0; в противном случае вывести номер первого недопустимого элемента.

PascalABC.NET

```

Var
  a: array of integer;
  N, i, i2: Integer;
  check: Boolean;
begin
  Write('N: ');
  Readln(N);

```

```

a := new integer[N + 1];
for i := 1 to N do
begin
  write(i, ' ');
  readln(a[i]);
end;
for i := 1 to N do
begin
  check := False;
  for i2 := 1 to N do
    if i = a[i2] then
      check := True;
  if not (check) then
  begin
    Writeln(i);
    break;
  end;
end;
if check then
  Writeln(0);
end.

```

C++

```

#include <iostream>
using namespace std;
int main() {
  int *x, i, j, n;
  cin >> n;
  x = new int[n];
  for (i = 0; i < n; ++i) {
    x[i] = rand() % 10;
    cout << x[i] << " ";
  }
  cout << endl;
  for (i = 1; i <= n; ++i) {
    for (j = 0; j < n; ++j) {
      if (x[j] == i) {
        j = n + 100;
        break;
      }
    }
    if (j != n + 100) {
      cout << i << endl;
      break;
    }
  }
  if (i == n + 1)
    cout << 0 << endl;
  system("pause");
  return 0;
}

```

3. Работа с несколькими массивами

12) Даны два массива *A* и *B* одинакового размера *N*. Сформировать новый массив *C* того же размера, каждый элемент которого равен максимальному из элементов массивов *A* и *B* с тем же индексом.

PascalABC.NET	C++
<pre> Var a, b, c: array of real; N, k: Integer; begin Write('N: '); Readln(N); a:=new integer[N+1]; b:=new integer[N+1]; c:=new integer[N+1]; Writeln('A: '); for k := 1 to N do begin write(k, ' :'); readln(a[k]); end; Writeln('B: '); for k := 1 to N do begin write(k, ' :'); readln(b[k]); end; for k := 1 to N do c[k] := max(a[k], b[k]); Writeln('C: '); for k := 1 to N do writeln(k, ' :', c[k]); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *x, *y, *z, i, j, n; cin >> n; x = new int[n]; y = new int[n]; z = new int[n]; for (i = 0; i < n; ++i) { x[i] = rand() % 50; y[i] = rand() % 60; cout << x[i] << " "; } cout << endl; for (i = 0; i < n; ++i) cout << y[i] << " "; for (int k = 0; k<n; ++k){ if (x[k]>y[k]) z[k] = x[k]; else z[k] = y[k]; } cout << endl; for (i = 0; i < n; ++i) cout << z[i] << " "; system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math/rand") func MaxInt32(a,b int) int{ if a>b{ return a }else{ return b } } func main() { var n int fmt.Print("n->") fmt.Scan(&n) var x=make([]int, n) var y=make([]int, n) </pre>	


```

var z=make([]int, n)
for i := 0; i < n; i++ {
    x[i]=rand.Intn(50);
    y[i]=rand.Intn(60);
    z[i]=MaxInt32(x[i],y[i]);
}
fmt.Println(x)
fmt.Println(y)
fmt.Println(z)
}

```

4. Преобразование массива

При выполнении заданий из данного пункта не следует использовать вспомогательные массивы.

а) Изменение элементов массива

13) Дан массив A размера N и целое число K ($1 \leq K \leq N$). Преобразовать массив, увеличив каждый его элемент на исходное значение элемента A_K .

PascalABC.NET	C++
<pre> Var a: array of integer; N, K, i, Ak: Integer; begin Write('N: '); Readln(N); a := new integer[N + 1]; Write('K: '); Readln(K); Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; Ak := a[k]; for i := 1 to N do a[i] := a[i] + Ak; for i := 1 to N do writeln(i, ' :', a[i]); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *x, i, k, n; cin >> n >> k; x = new int[n]; for (i = 0; i < n; ++i) { x[i] = rand() % 50; cout << x[i] << " "; } cout << endl; for (i = 0; i < n; ++i) { x[i] += x[k - 1]; } for (i = 0; i < n; ++i) cout << x[i] << " "; system("pause"); return 0; } </pre>

14) Дан массив размера N (N — четное число). Поменять местами первую и вторую половины массива.

PascalABC.NET	C++
<pre> Var a : array of integer; N,i:Integer; begin Write('N: '); Readln(N); </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n; cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { </pre>

<pre> a:=new integer[N+1]; Writeln('A: '); for i:=1 to N do begin readln(a[i]); end; i:=1; while i<=N/2 do begin swap(a[i],a[(N div 2)+i]); i:=i+1; end; for i:=1 to N do writeln(i, ' ',a[i]); end. </pre>	<pre> a[i] = rand() % 50; cout << a[i] << " "; } cout << endl; for (i = 0; i<n / 2; ++i) { a[i] += a[n / 2 + i]; a[n/2+i]=a[i]-a[n/2+i]; a[i]=a[i]-a[n / 2 + i]; } for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>
--	---

15) Дан массив размера N . Осуществить *сдвиг* элементов массива вправо на одну позицию (при этом A_1 перейдет в A_2 , A_2 — в A_3 , ..., A_{N-1} — в A_N , а исходное значение последнего элемента будет потеряно). Первый элемент полученного массива положить равным 0.

PascalABC.NET	C++
<pre> var a: array of Integer; N, i: Integer; begin Write('N: '); Readln(N); a := new integer[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' '); readln(a[i]); end; for i := N downto 2 do a[i] := a[i - 1]; A[1] := 0; for i := 1 to N do writeln(i, ':', a[i]); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n; cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 50; cout << a[i] << " "; } cout << endl; for (i = n - 1; i >= 1; --i) a[i] = a[i - 1]; a[0] = 0; for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>

16) Дан массив размера N , все элементы которого, кроме последнего, упорядочены по возрастанию. Сделать массив упорядоченным, переместив последний элемент на новую позицию.

PascalABC.NET	C++
<pre> Var a: array of Integer; N, i: Integer; begin Write('N: '); Readln(N); a := new integer[N + 1]; Writeln('A: '); </pre>	<pre> #include <iostream> using namespace std; void swap(float &a, float &b) { int h = a; a = b; b = h; } int main() { int *a, i, n; </pre>

<pre> for i := 1 to N do begin write(i, ' '); readln(a[i]); end; for i:=N-1 downto 1 do if a[i] > a[i + 1] then swap(a[i], a[i + 1]); for i := 1 to N do writeln(i, ':', a[i]); end. </pre>	<pre> cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 50; cout << a[i] << " "; } cout << endl; for (i = n - 2; i >= 0; --i) { if (a[i] > a[i + 1]) { swap(a[i], a[i + 1]); } else break; } for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>
--	---

b) Удаление и вставка элементов

17) Дан массив размера N и целое число K ($1 \leq K \leq N$). Удалить из массива элемент с порядковым номером K .

PascalABC.NET	C++
<pre> Var a: array of Integer; N, K, i: Integer; begin Write('N: '); Readln(N); a:=new integer[N + 1]; Write('K: '); Readln(K); Writeln('A: '); for i := 1 to N do begin write(i, ' '); readln(a[i]); end; N := N - 1; for i := K to N do a[i] := a[i + 1]; for i := 1 to N do writeln(i, ':', a[i]); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, k; cin >> n>>k; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 50; cout << a[i] << " "; } cout << endl; --n; for (i = k - 1; i < n; ++i) { a[i] = a[i + 1]; } for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>

18) Дан целочисленный массив размера N . Удалить из массива все нечетные числа и вывести размер полученного массива и его содержимое.

PascalABC.NET	C++
<pre> Var a: array of Integer; N, i, k: Integer; begin </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, k=0; </pre>

<pre> Readln(N); a := new integer[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; k := 0; for i := 1 to n do if not odd(a[i]) then begin inc(k); a[k] := a[i]; end; Writeln(N); for i := 1 to k do writeln(i, ' :', a[i]); end. </pre>	<pre> cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 50; cout << a[i] << " "; } cout << endl; for (i = 0; i < n; ++i) { if (a[i] % 2 == 0) { a[k] = a[i]; ++k; } } for (i = 0; i < k; ++i) cout << a[i] << " "; system("pause"); return 0; </pre>
--	---

19) Дан целочисленный массив размера N . Удалить из массива все элементы, встречающиеся менее трех раз, и вывести размер полученного массива и его содержимое.

PascalABC.NET	C++
<pre> Var a: array of Integer; N, i, i2, k, num: Integer; begin Write('N: '); Readln(N); a := new integer[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; repeat k := 11; for i := 1 to N do begin Num := 0; for i2 := 1 to N do if a[i]=a[i2] then inc(num); if num<3 then k := i; end; if k <> 11 then begin N := N - 1; for i := K to N do a[i] := a[i + 1]; end; </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, k = -1, num, j; cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 10; cout << a[i] << " "; } cout << endl; for (i = 0; i < n; ++i) { for (j = 0; j <= k; ++j) { if (a[i] == a[j]) break; } if (j != k + 1) { ++k; a[k] = a[i]; } else { num = 0; for (j = i; j < n; ++j) { if (a[j] == a[i]) ++num; } if (num >= 3) { ++k; a[k] = a[i]; } } } } </pre>

<pre> until (k = 11); Writeln(N); for i := 1 to N do writeln(i, ' :', a[i]); end. </pre>	<pre> } ++k; for (i = 0; i < k; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>
--	--

20) Дан массив размера N и целое число K ($1 \leq K \leq N$). Перед элементом массива с порядковым номером K вставить новый элемент с нулевым значением.

PascalABC.NET	C++
<pre> Var a: array of Integer; N, i, K: Integer; begin Write('N: '); Readln(N); a := new integer[N + 1]; Write('K: '); Readln(K); Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; N := N + 1; for i := N downto K + 1 do a[i] := a[i - 1]; a[k] := 0; Writeln(N); for i := 1 to N do writeln(i, ' :', a[i]); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, k; cin >> n >> k; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 10; cout << a[i] << " "; } cout << endl; n++; for (i = n - 1; i >= k; --i) a[i] = a[i - 1]; a[k - 1] = 0; for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>

21) Дан массив размера N . Перед каждым положительным элементом массива вставить элемент с нулевым значением.

PascalABC.NET	C++
<pre> var a, b: array of Integer; N, Nk, i, k: Integer; begin Write('N: '); Readln(N); a := new integer[N + 1]; b := new integer[2*N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; Nk := N; </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, j; cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 10-3; cout << a[i] << " "; } cout << endl; for (i = n - 1; i >= 0; --i) { if (a[i] > 0) { ++n; for (j = n - 1; j > i; --j) </pre>

<pre> k := 0; for i := 1 to N do begin if a[i] > 0 then begin Nk := Nk + 1; b[k + 1] := a[i]; b[k] := 0; K := k + 1; end else b[k] := a[i]; K := k + 1; end; end; N := Nk; a := b; Writeln(N); for i := 1 to N do writeln(i, ' :', a[i]); end. </pre>	<pre> a[j] = a[j - 1]; a[i] = 0; } } for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>
--	--

с) Сортировка массива

22) Дан массив A размера N (≤ 6). Упорядочить его по возрастанию методом сортировки *простым обменом* («*пузырьковой*» сортировкой): просматривать массив, сравнивая его соседние элементы (A_1 и A_2 , A_2 и A_3 и т. д.) и меняя их местами, если левый элемент пары больше правого; повторить описанные действия $N-1$ раз. Для контроля за выполняемыми действиями выводить содержимое массива после каждого просмотра. Учтеь, что при каждом просмотре количество анализируемых пар можно уменьшить на 1.

PascalABC.NET	C++
<pre> Var a: array of real; N, N2, i, i2, i3: integer; begin Write('N: '); Readln(N); a := new real[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; N2 := N; for i := 1 to N - 1 do begin i2 := i; N2 := N2 - 1; while (i2 <= N2) do begin if a[i2] > a[i2+1] then swap(a[i2], a[i2+1]); end; end; end; </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, j; cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 10 - 3; cout << a[i] << " "; } cout << endl; for (int i = 1; i < n; ++i) { for (int j = 0; j < n - i; j++) { if (a[j] < a[j + 1]) { int temp = a[j]; a[j] = a[j + 1]; a[j + 1] = temp; } } } } </pre>

<pre> inc(i2); end; for i3 := 1 to N do writeln(a[i3]:7:0); end; end. </pre>	<pre> for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>
--	---

23) Дан массив A размера N (≤ 6). Упорядочить его по возрастанию методом сортировки *простыми вставками*: сравнить элементы A_1 и A_2 и, при необходимости меняя их местами, добиться того, чтобы они оказались упорядоченными по возрастанию; затем обратиться к элементу A_3 и переместить его в левую (уже упорядоченную) часть массива, сохранив ее упорядоченность; повторить этот процесс для остальных элементов, выводя содержимое массива после обработки каждого элемента (от 2-го до N -го). При выполнении описанных действий удобно использовать прием «барьера», записывая очередной элемент перед его обработкой в дополнительный элемент массива A_0 .

PascalABC.NET	C++
<pre> var a: array of real; N, i, i2, i3: integer; begin Write('N: '); Readln(N); a := new real[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; if a[1] > a[2] then swap(a[1], a[2]); for i := 3 to N do begin a[0] := a[i]; for i2:=i-1 downto 0 do if a[0] < a[i2] then a[i2 + 1] := a[i2] else begin a[i2 + 1] := a[0]; break; end; for i3 := 1 to N do writeln(i3, ':', a[i3]); writeln; end; end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, j, temp; cin >> n; a = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 10 - 3; cout << a[i] << " "; } cout << endl; if (a[0] > a[1]) { temp = a[0]; a[0] = a[1]; a[1] = temp; } for (i = 2; i <= n - 1; ++i) { temp = a[i]; for (j = i - 1; j >= 0; --j) if (temp < a[j]) a[j+1]=a[j]; else break; a[j + 1] = temp; } for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>

5. Серии целых чисел

24) Дан целочисленный массив A размера N . Назовем *серией* группу подряд идущих одинаковых элементов, а *длиной серии* — количество этих элементов

(длина серии может быть равна 1). Сформировать два новых целочисленных массива *B* и *C* одинакового размера, записав в массив *B* длины всех серий исходного массива, а в массив *C* — значения элементов, образующих эти серии.

PascalABC.NET	C++
<pre> Var a, b, c: array of real; N, i, k: integer; begin Write('N: '); Readln(N); a:=new real[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; k := 1; b[k] := 1; c[k] := a[k]; for i := 2 to N do begin if a[i-1]=a[i] then b[k] := b[k] + 1 else begin inc(k); b[k] := 1; c[k] := a[i]; end; end; for i := 1 to k do writeln(c[i], ' - ', b[i]); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int *a,*b,*c, i, n, j, k=0; cin >> n; a = new int[n]; b = new int[n]; c = new int[n]; for (i = 0; i < n; ++i) { a[i] = rand() % 5; cout << a[i] << " "; } cout << endl; b[k] = 1; c[k] = a[k]; for (i = 1; i < n; ++i) { if (a[i - 1] == a[i]) ++b[k]; else { ++k; b[k] = 1; c[k] = a[i]; } } for (i = 0; i < k; ++i) cout <<b[i] << " "; cout << endl; for (i = 0; i < k; ++i) cout << c[i] << " "; system("pause"); return 0; } </pre>

25) Дан целочисленный массив размера *N*. Вставить перед каждой его серией элемент с нулевым значением.

PascalABC.NET	C++
<pre> Var a, b: array of real; N, i, k: integer; begin Write('N: '); Readln(N); a:=new real[N + 1]; b:=new real[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; b[1] := 0; </pre>	<pre> #include <iostream> using namespace std; int main() { int *a,*b, i, n, k=1; cin >> n; a = new int[n]; b = new int[n*2]; for (i = 0; i < n; ++i) { a[i] = rand() % 5; cout << a[i] << " "; } cout << endl; b[0] = 0; b[k] = a[1]; for (i = 1; i<n; ++i) { </pre>

<pre> k := 2; b[k] := a[1]; for i := 2 to N do begin inc(k); if a[i-1] <> a[i] then begin b[k] := 0; inc(k); end; b[k] := a[i]; end; a := b; for i := 1 to k do writeln(a[i]); end. </pre>	<pre> ++k; if (a[i - 1] != a[i]) { b[k] = 0; ++k; } b[k] = a[i]; } for (i = 0; i < k; ++i) cout <<b[i] << " "; system("pause"); return 0; } </pre>
--	---

26) Дано целое число $K (> 0)$ и целочисленный массив размера N . Преобразовать массив, удвоив длину его серии с номером K . Если серий в массиве меньше K , то вывести массив без изменений.

PascalABC.NET	C++
<pre> Var a, b: array of real; N, K, i, i2, ki: integer; begin Write('N: '); Readln(N); a := new real[N + 1]; b := new real[N + 1]; Write('K: '); Readln(K); Writeln('A: '); for i := 1 to N do begin write(i, ' :'); readln(a[i]); end; i2 := 0; ki := 1; for i := 1 to N - 1 do begin if ki = K then begin inc(i2); b[i2] := a[i]; end; inc(i2); b[i2] := a[i]; if a[i] <> a[i + 1] then inc(ki); end; if ki = K then begin inc(i2); b[i2] := a[i]; end; </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, k = 1; cin >> n; a = new int[2 * n]; for (i = 0; i < n; ++i) { a[i] = rand() % 5; cout << a[i] << " "; } cout << endl; int num = 1; for (i=0;(i<n-1)&&(num<k);++i) { if (a[i] != a[i + 1]) ++num; } int begine = i; for (i=begine; (i<n-1); ++i) { if (a[i] != a[i + 1]) break; } int ende = i; int i2; for (i2=1;i2<=ende-begine+1; ++i2) { n++; for (i = n - 1; i > ende; i--) { a[i] = a[i - 1]; } } for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>

<pre> end; inc(i2); b[i2] := a[i]; a := b; for i := 1 to i2 do writeln(a[i]); end. </pre>	
---	--

27) Дан целочисленный массив размера N . Преобразовать массив, увеличив его последнюю серию наибольшей длины на один элемент.

PascalABC.NET	C++
<pre> Const col = 20; var a, c, temp: array [1..col] of real; x, y: array [1..col] of integer; N, i, i2, i3, ns, ms: integer; begin Write('N: '); Readln(N); Writeln('A: '); for i := 1 to N do begin write(i, ' '); readln(a[i]); end; ns := 1; x[ns] := 1; y[ns] := ns; ms := ns; c[ns] := a[1]; for i := 2 to N do begin if a[i - 1] = a[i] then begin x[ns] := x[ns] + 1 end else begin if x[ns] >= x[ms] then ms := ns; inc(ns); x[ns] := 1; y[ns] := i; c[ns] := a[i]; end; end; end; if x[ns] >= x[ms] then ms := ns; y[ns + 1] := N + 1; x[ms] := x[ms] + 1; i3 := 0; </pre>	<pre> #include <iostream> using namespace std; int main() { int *a, i, n, k = 1; cin >> n; a = new int[2 * n]; for (i = 0; i < n; ++i) { a[i] = rand() % 5; cout << a[i] << " "; } cout << endl; int len = 1, maxlen = 1, endmaxseries = 1; for (i = 1; i < n; ++i) { if (a[i - 1] != a[i]) { if (len >= maxlen) { maxlen = len; endmaxseries = i - 1; } len = 0; } ++len; } if (len > maxlen) { maxlen = len; endmaxseries = i - 1; } for (i = ++n - 1; i > endmaxseries; --i) a[i] = a[i - 1]; for (i = 0; i < n; ++i) cout << a[i] << " "; system("pause"); return 0; } </pre>

<pre> for i := 1 to ns do for i2 := 1 to x[i] do begin inc(i3); temp[i3] := c[i]; end; a := temp; for i := 1 to i3 do writeln(a[i]); end. </pre>	
---	--

6. Множества точек на плоскости

Для хранения данных о каждом наборе точек следует использовать по два массива: первый массив для хранения абсцисс, второй — для хранения ординат. Можно также использовать массив *записей* с двумя полями.

28) Дано множество A из N точек на плоскости и точка B (точки заданы своими координатами x, y). Найти точку из множества A , наиболее близкую к точке B . Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле: $R = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

PascalABC.NET
<pre> Var Ax, Ay: array of real; Bx, By, R, Rmin: real; N, i, Amin: integer; begin Write('N: '); Readln(N); Ax:=new real[N + 1]; Ay:=new real[N + 1]; Writeln('A: '); for i := 1 to N do begin write(i, '.x :'); readln(Ax[i]); write(i, '.y :'); readln(Ay[i]); end; Writeln('B'); Write('x: '); Readln(Bx); Write('y: '); Readln(By); Rmin := sqrt(sqr(AX[1] - Bx) + sqr(Ay[1] - By)); Amin := 1; for i := 2 to N do begin R := sqrt(sqr(Ax[i] - Bx) + sqr(Ay[i] - By)); if R < Rmin then begin </pre>

```

    Rmin := R;
    Amin := i;
  end;
end;
Writeln(Amin);
end.

```

C++

```

#include <iostream>
using namespace std;
int main() {
    float a[100][2];
    int n, amin = INT32_MAX;
    float bx, by;
    cin >> n >> bx >> by;
    int i;
    for (i = 0; i < n; ++i) {
        cout << (i + 1) << endl;
        cout << "x=";
        cin >> a[i][0];
        cout << "y=";
        cin >> a[i][1];
    }
    float r, rmin=32875;
    for (i = 0; i < n; ++i) {
        r=sqrt(pow(bx-a[i][0],2)+pow(by- a[i][1], 2));
        if (r < rmin || i == 0) {
            rmin = r;
            amin = i;
        }
    }
    cout<<amin+1<<" "<< a[amin][0]
        << " " << a[amin][1];
    system("pause");
    return 0;
}

```

- 29) Даны множества A и B , состоящие соответственно из N_1 и N_2 точек (точки заданы своими координатами x , y). Найти минимальное расстояние между точками этих множеств и сами точки, расположенные на этом расстоянии (вначале выводится точка из множества A , затем точка из множества B).

PascalABC.NET

```

Var
Ax,Ay: array of real;
R,Rmax:real;
N,i,i2,Point1,Point2: integer;

begin
Write('N: ');
Readln(N);
Ax:=new real[N+1];
Ay:=new real[N+1];
Writeln('A: ');
for i:=1 to N do
    begin

```

```

write(i, '.x :');
readln(ax[i]);
write(i, '.y :');
readln(Ay[i]);
end;
Rmax:=0;
Point1:=0;
Point2:=0;
for i:=1 to N-1 do
  for i2:=i+1 to N do
    begin
      R:=sqrt(sqr(Ax[i]-Ay[i2])+sqr(Ax[i]-Ay[i2]));
      writeln(R);
      if (R > Rmax) then
        begin
          Rmax:=R;
          Point1:=i;
          Point2:=i2;
        end;
      end;
    end;

  Writeln ('(',Ax[Point1],':',Ay[Point1],')(',Ax[Point2],':',
          Ay[Point2],')');
end.

```

C++

```

#include <iostream>
using namespace std;
int main() {
    float a[100][2];
    int i, n;
    cin >> n;
    for (i = 0; i < n; ++i)
        cin >> a[i][0] >> a[i][1];
    int a1 = 0, a2 = 0, i2;
    float r, rmax = 0;
    for (i = 0; i < n; ++i) {
        for (i2 = i + 1; i2 < n; ++i2) {
            r=sqrt(pow(a[i][0]-a[i2][0],2)+pow(a[i][1]-a[i2][1],2));
            if (r > rmax) {
                rmax = r;
                a1 = i;
                a2 = i2;
            }
        }
    }
    cout << a1 + 1 << " " << a[a1][0] << " " << a[a1][1];
    cout << a2 + 1 << " " << a[a2][0] << " " << a[a2][1];
    system("pause");
    return 0;
}

```

Задачи для самостоятельной работы

Формирование массива и вывод его элементов

- 1) Дано целое число $N (> 0)$. Сформировать и вывести целочисленный массив размера N , содержащий степени двойки от первой до N -й: 2, 4, 8, 16,
- 2) Дано целое число $N (> 1)$, а также первый член A и знаменатель Q геометрической прогрессии. Сформировать и вывести массив размера N , содержащий N первых членов данной прогрессии:

$$A, A \cdot D, A \cdot D^2, A \cdot D^3, \dots$$

- 3) Дано целое число $N (> 2)$. Сформировать и вывести целочисленный массив размера N , содержащий N первых элементов последовательности чисел Фибоначчи F_K :

$$F_1 = 1, \quad F_2 = 1, \quad F_K = F_{K-2} + F_{K-1}, \quad K = 3, 4, \dots$$

- 4) Даны целые числа $N (> 2)$, A и B . Сформировать и вывести целочисленный массив размера N , первый элемент которого равен A , второй равен B , а каждый последующий элемент равен сумме всех предыдущих.
- 5) Дан массив размера N . Вывести его элементы в обратном порядке.
- 6) Дан целочисленный массив размера N . Вывести все содержащиеся в данном массиве четные числа в порядке убывания их индексов, а также их количество K .
- 7) Дан целочисленный массив размера N . Вывести вначале все содержащиеся в данном массиве четные числа в порядке возрастания их индексов, а затем — все нечетные числа в порядке убывания их индексов.
- 8) Дан массив A размера N (N — четное число). Вывести его элементы с четными номерами в порядке возрастания номеров: $A_2, A_4, A_6, \dots, A_N$. Условный оператор не использовать.
- 9) Дан массив A размера N (N — нечетное число). Вывести его элементы с нечетными номерами в порядке убывания номеров: $A_N, A_{N-2}, A_{N-4}, \dots, A_1$. Условный оператор не использовать.
- 10) Дан массив A размера N . Вывести вначале его элементы с четными номерами (в порядке возрастания номеров), а затем — элементы с нечетными номерами (также в порядке возрастания номеров):

$$A_2, A_4, A_6, \dots, A_1, A_3, A_5, \dots$$

Условный оператор не использовать.

- 11) Дан массив A размера N . Вывести вначале его элементы с нечетными номерами в порядке возрастания номеров, а затем — элементы с четными номерами в порядке убывания номеров.

$$A_1, A_3, A_5, \dots, A_6, A_4, A_2.$$

Условный оператор не использовать.

- 12) Дан массив A размера N . Вывести его элементы в следующем порядке:

$$A_1, A_N, A_2, A_{N-1}, A_3, A_{N-2}, \dots$$

- 13) Дан массив A размера N . Вывести его элементы в следующем порядке:

$$A_1, A_2, A_N, A_{N-1}, A_3, A_4, A_{N-2}, A_{N-3}, \dots$$

Анализ элементов массива

- 14) Дан целочисленный массив A размера 10. Вывести порядковый номер последнего из тех его элементов A_K , которые удовлетворяют двойному неравенству $A_1 < A_K < A_{10}$. Если таких элементов нет, то вывести 0.

- 15) Дан массив размера N и целые числа K и L ($1 \leq K \leq L \leq N$). Найти сумму элементов массива с номерами от K до L включительно.
- 16) Дан массив размера N и целые числа K и L ($1 \leq K \leq L \leq N$). Найти среднее арифметическое элементов массива с номерами от K до L включительно.
- 17) Дан массив размера N и целые числа K и L ($1 < K \leq L \leq N$). Найти сумму всех элементов массива, кроме элементов с номерами от K до L включительно.
- 18) Дан массив размера N и целые числа K и L ($1 < K \leq L \leq N$). Найти среднее арифметическое всех элементов массива, кроме элементов с номерами от K до L включительно.
- 19) Дан целочисленный массив размера N , не содержащий одинаковых чисел. Проверить, образуют ли его элементы *арифметическую прогрессию*. Если образуют, то вывести разность прогрессии, если нет — вывести 0.
- 20) Дан массив ненулевых целых чисел размера N . Проверить, образуют ли его элементы *геометрическую прогрессию*. Если образуют, то вывести знаменатель прогрессии, если нет — вывести 0.
- 21) Дан целочисленный массив размера N . Проверить, чередуются ли в нем четные и нечетные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
- 22) Дан массив ненулевых целых чисел размера N . Проверить, чередуются ли в нем положительные и отрицательные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
- 23) Дан массив A размера N . Найти максимальный элемент из его элементов с нечетными номерами: A_1, A_3, A_5, \dots .
- 24) Дан массив размера N . Найти номера тех элементов массива, которые больше своего правого соседа, и количество таких элементов. Найденные номера выводить в порядке их возрастания.
- 25) Дан массив размера N . Найти номера тех элементов массива, которые больше своего левого соседа, и количество таких элементов. Найденные номера выводить в порядке их убывания.
- 26) Дан массив размера N . Найти номер его последнего локального максимума (*локальный максимум* — это элемент, который больше любого из своих соседей).
- 27) Дан массив размера N . Найти максимальный из его локальных минимумов.
- 28) Дан массив размера N . Найти минимальный из его локальных максимумов.
- 29) Дан массив размера N . Найти максимальный из его элементов, не являющихся ни локальным минимумом, ни локальным максимумом. Если таких элементов в массиве нет, то вывести 0.
- 30) Дан массив размера N . Найти количество участков, на которых его элементы монотонно убывают.
- 31) Дан массив размера N . Найти количество его *промежутков монотонности* (то есть участков, на которых его элементы возрастают или убывают).
- 32) Дано число R и массив A размера N . Найти элемент массива, который *наиболее близок* к числу R (то есть такой элемент A_K , для которого величина $|A_K - R|$ является минимальной).
- 33) Дан массив размера N . Найти два соседних элемента, сумма которых максимальна, и вывести эти элементы в порядке возрастания их индексов.

- 34) Дано число R и массив размера N . Найти два соседних элемента массива, сумма которых наиболее близка к числу R , и вывести эти элементы в порядке возрастания их индексов.
- 35) Дан целочисленный массив размера N , содержащий ровно два одинаковых элемента. Найти номера одинаковых элементов и вывести эти номера в порядке возрастания.
- 36) Дан массив размера N . Найти номера двух ближайших элементов из этого массива (то есть элементов с наименьшим модулем разности) и вывести эти номера в порядке возрастания.
- 37) Дан целочисленный массив размера N . Найти количество различных элементов в данном массиве.
- 38) Дан целочисленный массив размера N . Найти максимальное количество его одинаковых элементов.
- 39) Дан целочисленный массив A размера N , являющийся перестановкой. Найти количество *инверсий* в данной перестановке, то есть таких пар элементов A_I и A_J , в которых большее число находится слева от меньшего: $A_I > A_J$ при $I < J$.
- 40) Даны массивы A и B одинакового размера N . Поменять местами их содержимое и вывести вначале элементы преобразованного массива A , а затем — элементы преобразованного массива B .

Работа с несколькими массивами

- 41) Дан массив A размера N . Сформировать новый массив B того же размера, элементы которого определяются следующим образом:

$$B_k = \begin{cases} 2 \cdot A_k, & \text{если } A_k < 5, \\ \frac{A_k}{2} & \text{в противном случае.} \end{cases}$$

- 42) Дан целочисленный массив A размера N . Переписать в новый целочисленный массив B все четные числа из исходного массива (в том же порядке) и вывести размер полученного массива B и его содержимое.
- 43) Дан целочисленный массив A размера N (≤ 15). Переписать в новый целочисленный массив B все элементы с нечетными порядковыми номерами (1, 3, ...) и вывести размер полученного массива B и его содержимое. Условный оператор не использовать.
- 44) Дан целочисленный массив A размера N (≤ 15). Переписать в новый целочисленный массив B все элементы с порядковыми номерами, кратными трем (3, 6, ...), и вывести размер полученного массива B и его содержимое. Условный оператор не использовать.
- 45) Дан целочисленный массив A размера N . Переписать в новый целочисленный массив B того же размера вначале все элементы исходного массива с четными номерами, а затем — с нечетными:

$$A_2, A_4, A_6, \dots, A_1, A_3, A_5, \dots$$

Условный оператор не использовать.

- 46) Дан массив A размера N . Сформировать новый массив B того же размера по следующему правилу: элемент B_K равен сумме элементов массива A с номерами от 1 до K .

- 47) Дан массив A размера N . Сформировать новый массив B того же размера по следующему правилу: элемент B_K равен среднему арифметическому элементов массива A с номерами от 1 до K .
- 48) Дан массив A размера N . Сформировать новый массив B того же размера по следующему правилу: элемент B_K равен сумме элементов массива A с номерами от K до N .
- 49) Дан массив A размера N . Сформировать новый массив B того же размера по следующему правилу: элемент B_K равен среднему арифметическому элементов массива A с номерами от K до N .
- 50) Дан массив A размера N . Сформировать два новых массива B и C : в массив B записать все положительные элементы массива A , в массив C — все отрицательные (сохраняя исходный порядок следования элементов). Вывести вначале размер и содержимое массива B , а затем — размер и содержимое массива C .
- 51) Даны два массива A и B размера 5, элементы которых упорядочены по возрастанию. Объединить эти массивы так, чтобы результирующий массив C (размера 10) остался упорядоченным по возрастанию.
- 52) Даны три целочисленных массива A , B и C размера N_A , N_B , N_C соответственно, элементы которых упорядочены по убыванию. Объединить эти массивы так, чтобы результирующий целочисленный массив D (размера $N_A + N_B + N_C$) остался упорядоченным по убыванию.

Преобразование массива

При выполнении заданий из данного пункта не следует использовать вспомогательные массивы.

Изменение элементов массива

- 53) Дан целочисленный массив размера N . Увеличить все четные числа, содержащиеся в массиве, на исходное значение первого четного числа. Если четные числа в массиве отсутствуют, то оставить массив без изменений.
- 54) Дан целочисленный массив размера N . Увеличить все нечетные числа, содержащиеся в массиве, на исходное значение последнего нечетного числа. Если нечетные числа в массиве отсутствуют, то оставить массив без изменений.
- 55) Дан массив размера N . Поменять местами его минимальный и максимальный элементы.
- 56) Дан массив размера N (N — четное число). Поменять местами его первый элемент со вторым, третий — с четвертым и т. д.
- 57) Дан массив размера N . Поменять порядок его элементов на обратный.
- 58) Дан массив A размера N и целые числа K и L ($1 \leq K < L \leq N$). Переставить в обратном порядке элементы массива, расположенные между элементами A_K и A_L , включая эти элементы.
- 59) Дан массив A размера N и целые числа K и L ($1 \leq K < L \leq N$). Переставить в обратном порядке элементы массива, расположенные между элементами A_K и A_L , не включая эти элементы.
- 60) Дан массив размера N . Обнулить элементы массива, расположенные между его минимальным и максимальным элементами (не включая минимальный и максимальный элементы).

- 61) Дан массив размера N . Переставить в обратном порядке элементы массива, расположенные между его минимальным и максимальным элементами, включая минимальный и максимальный элементы.
- 62) Дан массив размера N . Обнулить все его *локальные максимумы* (то есть числа, большие своих соседей).
- 63) Дан массив размера N . Возвести в квадрат все его *локальные минимумы* (то есть числа, меньшие своих соседей).
- 64) Дан массив размера N . Заменить каждый элемент массива на среднее арифметическое этого элемента и его соседей.
- 65) Дан массив размера N . Осуществить *сдвиг* элементов массива влево на одну позицию (при этом A_N перейдет в A_{N-1} , A_{N-1} — в A_{N-2} , ..., A_2 — в A_1 , а исходное значение первого элемента будет потеряно). Последний элемент полученного массива положить равным 0.
- 66) Дан массив размера N и целое число K ($1 \leq K < N$). Осуществить *сдвиг* элементов массива вправо на K позиций (при этом A_1 перейдет в A_{K+1} , A_2 — в A_{K+2} , ..., A_{N-K} — в A_N , а исходное значение K последних элементов будет потеряно). Первые K элементов полученного массива положить равными 0.
- 67) Дан массив размера N и целое число K ($1 \leq K < N$). Осуществить *сдвиг* элементов массива влево на K позиций (при этом A_N перейдет в A_{N-K} , A_{N-1} — в A_{N-K-1} , ..., A_{K+1} — в A_1 , а исходное значение K первых элементов будет потеряно). Последние K элементов полученного массива положить равными 0.
- 68) Дан массив размера N . Осуществить *циклический сдвиг* элементов массива вправо на одну позицию (при этом A_1 перейдет в A_2 , A_2 — в A_3 , ..., A_N — в A_1).
- 69) Дан массив размера N . Осуществить *циклический сдвиг* элементов массива влево на одну позицию (при этом A_N перейдет в A_{N-1} , A_{N-1} — в A_{N-2} , ..., A_1 — в A_N).
- 70) Дан массив A размера N и целое число K ($1 \leq K \leq 4$, $K < N$). Осуществить *циклический сдвиг* элементов массива вправо на K позиций (при этом A_1 перейдет в A_{K+1} , A_2 — в A_{K+2} , ..., A_N — в A_K). Допускается использовать вспомогательный массив из 4 элементов.
- 71) Дан массив A размера N и целое число K ($1 \leq K \leq 4$, $K < N$). Осуществить *циклический сдвиг* элементов массива влево на K позиций (при этом A_N перейдет в A_{N-K} , A_{N-1} — в A_{N-K-1} , ..., A_1 — в A_{N-K+1}). Допускается использовать вспомогательный массив из 4 элементов.
- 72) Дан массив размера N , все элементы которого, кроме первого, упорядочены по возрастанию. Сделать массив упорядоченным, переместив первый элемент на новую позицию.
- 73) Дан массив размера N , все элементы которого, кроме последнего, упорядочены по возрастанию. Сделать массив упорядоченным, переместив последний элемент на новую позицию.
- 74) Дан массив размера N , все элементы которого, кроме одного, упорядочены по убыванию. Сделать массив упорядоченным, переместив элемент, нарушающий упорядоченность, на новую позицию.

Удаление и вставка элементов

- 75) Дан массив размера N и целые числа K и L ($1 \leq K < L \leq N$). Удалить из массива элементы с номерами от K до L включительно и вывести размер полученного массива и его содержимое.
- 76) Дан целочисленный массив размера N (> 2). Удалить из массива все элементы с четными номерами (2, 4, ...). Условный оператор не использовать.
- 77) Дан целочисленный массив размера N (> 2). Удалить из массива все элементы с нечетными номерами (1, 3, ...). Условный оператор не использовать.
- 78) Дан целочисленный массив размера N . Удалить из массива все соседние одинаковые элементы, оставив их первые вхождения.
- 79) Дан целочисленный массив размера N . Удалить из массива все одинаковые элементы, оставив их первые вхождения.
- 80) Дан целочисленный массив размера N . Удалить из массива все одинаковые элементы, оставив их последние вхождения.
- 81) Дан целочисленный массив размера N . Удалить из массива все элементы, встречающиеся более двух раз, и вывести размер полученного массива и его содержимое.
- 82) Дан целочисленный массив размера N . Удалить из массива все элементы, встречающиеся ровно два раза, и вывести размер полученного массива и его содержимое.
- 83) Дан массив размера N и целое число K ($1 \leq K \leq N$). После элемента массива с порядковым номером K вставить новый элемент с нулевым значением.
- 84) Дан массив размера N . Вставить элемент с нулевым значением перед минимальным и после максимального элемента массива.
- 85) Дан массив размера N и два целых числа K и M ($1 \leq K \leq N$, $1 \leq M \leq 10$). Перед элементом массива с номером K вставить M новых элементов с нулевыми значениями.
- 86) Дан массив размера N и два целых числа K и M ($1 \leq K \leq N$, $1 \leq M \leq 10$). После элемента массива с номером K вставить M новых элементов с нулевыми значениями.
- 87) Дан массив размера N . Продублировать в нем элементы с четными номерами (2, 4, ...). Условный оператор не использовать.
- 88) Дан массив размера N . Утроить в нем вхождения всех элементов с нечетными номерами (1, 3, ...). Условный оператор не использовать.
- 89) Дан массив размера N . После каждого отрицательного элемента массива вставить элемент с нулевым значением.
- 90) Дан целочисленный массив размера N . Продублировать в нем все четные числа.
- 91) Дан целочисленный массив размера N . Утроить в нем вхождения всех нечетных чисел.

Сортировка массива

- 92) Дан массив A размера N (≤ 6). Упорядочить его по возрастанию методом сортировки *простым выбором*: найти максимальный элемент массива и поменять его местами с последним (N -м) элементом; выполнить описанные действия $N - 1$ раз, каждый раз уменьшая на 1 количество анализируемых элементов и выводя содержимое массива.

93) Дан массив A размера N . Не изменяя данный массив, вывести номера его элементов в том порядке, в котором соответствующие им элементы образуют возрастающую последовательность. Использовать метод «пузырьковой» сортировки, модифицировав его следующим образом: создать вспомогательный целочисленный массив номеров I , заполнив его числами от 1 до N ; просматривать массив A , сравнивая пары элементов массива A с номерами I_1 и I_2 , I_2 и I_3 , ... и меняя местами соответствующие элементы массива I , если левый элемент пары больше правого. Повторив описанную процедуру просмотра $N - 1$ раз, получим в массиве I требуемую последовательность номеров.

Серии целых чисел

- 94) Дан целочисленный массив размера N . Вставить после каждой его серии элемент с нулевым значением.
- 95) Дан целочисленный массив размера N . Преобразовать массив, увеличив каждую его серию на один элемент.
- 96) Дан целочисленный массив размера N , содержащий по крайней мере одну серию, длина которой больше 1. Преобразовать массив, уменьшив каждую его серию на один элемент.
- 97) Дано целое число $K (> 1)$ и целочисленный массив размера N . Удалить из массива серию с номером K . Если серий в массиве меньше K , то вывести массив без изменений.
- 98) Дано целое число $K (> 1)$ и целочисленный массив размера N . Поменять местами первую серию массива и его серию с номером K . Если серий в массиве меньше K , то вывести массив без изменений.
- 99) Дано целое число $K (> 0)$ и целочисленный массив размера N . Поменять местами последнюю серию массива и его серию с номером K . Если серий в массиве меньше K , то вывести массив без изменений.
- 100) Дано целое число $L (> 1)$ и целочисленный массив размера N . Заменить каждую серию массива, длина которой меньше L , на один элемент с нулевым значением.
- 101) Дано целое число $L (> 0)$ и целочисленный массив размера N . Заменить каждую серию массива, длина которой равна L , на один элемент с нулевым значением.
- 102) Дано целое число $L (> 0)$ и целочисленный массив размера N . Заменить каждую серию массива, длина которой больше L , на один элемент с нулевым значением.
- 103) Дан целочисленный массив размера N . Преобразовать массив, увеличив его первую серию наибольшей длины на один элемент.
- 104) Дан целочисленный массив размера N . Преобразовать массив, увеличив все его серии наибольшей длины на один элемент.

Множества точек на плоскости

Для хранения данных о каждом наборе точек следует использовать по два массива: первый массив для хранения абсцисс, второй — для хранения ординат. Можно также использовать массив *записей* с двумя полями.

- 105) Дано множество A из N точек (точки заданы своими координатами x , y). Среди всех точек этого множества, лежащих во второй четверти, найти

- точку, наиболее удаленную от начала координат. Если таких точек нет, то вывести точку с нулевыми координатами.
- 106) Дано множество A из N точек (точки заданы своими координатами x, y). Среди всех точек этого множества, лежащих в первой или третьей четверти, найти точку, наиболее близкую к началу координат. Если таких точек нет, то вывести точку с нулевыми координатами.
- 107) Дано множество A из N точек (точки заданы своими координатами x, y). Найти пару различных точек этого множества с максимальным расстоянием между ними и само это расстояние (точки выводятся в том же порядке, в котором они перечислены при задании множества A).
- 108) Дано множество A из N точек ($N > 2$, точки заданы своими координатами x, y). Найти такую точку из данного множества, сумма расстояний от которой до остальных его точек минимальна, и саму эту сумму.
- 109) Дано множество A из N точек ($N > 2$, точки заданы своими координатами x, y). Найти наибольший периметр треугольника, вершины которого принадлежат различным точкам множества A , и сами эти точки (точки выводятся в том же порядке, в котором они перечислены при задании множества A).
- 110) Дано множество A из N точек ($N > 2$, точки заданы своими координатами x, y). Найти наименьший периметр треугольника, вершины которого принадлежат различным точкам множества A , и сами эти точки (точки выводятся в том же порядке, в котором они перечислены при задании множества A).
- 111) Дано множество A из N точек с целочисленными координатами x, y . *Порядок* на координатной плоскости определим следующим образом:
 $(x_1, y_1) < (x_2, y_2)$, если либо $x_1 < x_2$, либо $x_1 = x_2$ и $y_1 < y_2$. Расположить точки данного множества по возрастанию в соответствии с указанным порядком.
- 112) Дано множество A из N точек с целочисленными координатами x, y . *Порядок* на координатной плоскости определим следующим образом:
 $(x_1, y_1) < (x_2, y_2)$, если либо $x_1 + y_1 < x_2 + y_2$, либо $x_1 + y_1 = x_2 + y_2$ и $x_1 < x_2$.
 Расположить точки данного множества по убыванию в соответствии с указанным порядком.

12. Двумерные массивы (матрицы): группа Matrix

Условие вида «дана матрица размера $M \times N$ » означает, что вначале дается *фактический размер* двумерного массива-матрицы (количество строк M и количество столбцов N), а затем приводятся элементы этого массива (количество элементов равно $M \cdot N$). Если в задании явно не указывается, какие значения могут принимать размеры исходной матрицы, то предполагается, что и число строк, и число столбцов может меняться в пределах от 2 до 10. Порядковые номера начальной строки и начального столбца матрицы считаются равными 1. Ввод и вывод элементов матрицы осуществляются *по строкам*.

Квадратной матрицей порядка M называется двумерный массив-матрица размера M .

Если в задании, связанном с созданием (преобразованием) матрицы, не описан результирующий набор данных, то предполагается, что этим набором является созданная (преобразованная) матрица, и необходимо вывести все ее элементы.

1. Формирование матрицы и вывод ее элементов

В заданиях на формирование матрицы предполагается, что размер результирующей матрицы не превосходит 10.

- 1) Даны целые положительные числа M и N . Сформировать целочисленную матрицу размера $M \times N$, у которой все элементы I -й строки имеют значение $10 \cdot I$ ($I = 1, \dots, M$).

PascalABC.NET	C++
<pre> Var i, j: Integer; begin var m := ReadInteger('m'); var n:= ReadInteger('n: '); var a: array [,] of integer :=new integer[m + 1, n + 1]; for i := 1 to M do for j := 1 to N do a[i, j] := i * 10; for i := 1 to M do begin for j := 1 to N do Write(a[i, j], ''); writeln(); end; end. </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, n, m, i, j; cin >> m >> n; x = new int*[m]; for (i = 0; i < m; i++) x[i] = new int[n]; for (i = 0; i < m; i++) for (j = 0; j < n; j++) x[i][j] = 10 * (i+1); for (i = 0; i < m; i++) { for (j = 0; j < n; j++) cout<<x[i][j] << " "; cout << endl; } system("pause"); return 0; } </pre>
Python	C#
<pre> from random import random a=[] n=int(input("")) for i in range(1,n+1): x=[] for j in range (1,n+1): x.append(i*10) </pre>	<pre> static void Main(string[] args) { int m, n, i, j; m=Convert.ToInt32(Console.ReadLine()); n = Convert.ToInt32(Console.ReadLine()); int [,]x=new Int32[m,n]; for (i=0; i<m; i++) for (j=0; j<n; j++) </pre>

<pre>a.append(x); for row in a: print(row)</pre>	<pre>x[i, j]=(i+1)*10; for (i=0; i<m; i++) { for (j=0; j<n; j++) Console.Write(x[i, j]+" "); Console.WriteLine(); } Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args){ Scanner sc=new Scanner(System.in); int m, n, i, j; m=sc.nextInt(); n=sc.nextInt(); int [][] x=new int[m][n]; Random rnd=new Random(); for (i=0; i<m; i++) for (j=0; j<n; j++) x[i][j]=rnd.nextInt(50); for (i=0; i<m; i++){ for (j=0; j<n; j++) System.out.print(x[i][j]+" "); System.out.println(); } }</pre>	<pre>package main import "fmt" func main() { var n,m int x := [10][10]int{} fmt.Print("n, m(<10)") fmt.Scan(&n, &m) for i:=0; i<n; i++ { for j:=0; j<m; j++ { x[i][j] = 10*(i+1) } } for i:=0; i<n; i++ { for j:=0; j<m; j++ { fmt.Printf("%v ",x[i][j]) } fmt.Println() } }</pre>
Kotlin	
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var m = reader.nextInt() var n = reader.nextInt() val a = Array(m) { arrayOfNulls<Int>(n) } for (i in (0..m - 1)) { for (j in (0..n - 1)) a[i][j] = 10 * (i + 1) } for (i in (0..m - 1)) { for (j in (0..n - 1)) print(a[i][j].toString() + " ") println() } }</pre>	
VB.NET	
<pre>Sub Main() Dim a(,) As Integer Dim i, j, m, n As Integer m = Convert.ToInt32(Console.ReadLine()) n = Convert.ToInt32(Console.ReadLine())</pre>	

```

a = New Integer(m, n) {}
For i = 0 To m - 1
    For j = 0 To n - 1
        a(i, j) = 10 * (i + 1)
    Next
Next
For i = 0 To m - 1
    For j = 0 To n - 1
        Console.Write("{0} ", a(i, j))
    Next
    Console.WriteLine()
Next
Console.ReadKey()
End Sub

```

2) Даны целые положительные числа M , N и набор из M чисел. Сформировать матрицу размера $M \times N$, у которой в каждом столбце содержатся все числа из исходного набора (в том же порядке).

PascalABC.NET	C++
<pre> Var a: array [1..10, 1..10] of integer; M, N, i, j: Integer; begin Write('N,M: '); Readln(N, M); Writeln('Nabor:'); for i := 1 to M do begin Write(i, ': '); readln(a[i, 1]); end; for i := 1 to M do for j := 2 to N do a[i, j] := a[i, j - 1]; for i := 1 to M do begin for j := 1 to N do Write(a[i, j], ' '); writeln; end; end. </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, n, m, i, j; cin >> m >> n; x = new int*[m]; for (i = 0; i < m; i++) x[i] = new int[n]; for (i = 0; i < m; i++) cin >> x[i][0]; for (i = 0; i < m; i++) for (j = 1; j < n; j++) x[i][j] = x[i][j - 1]; for (i = 0; i < m; i++) { for (j = 0; j < n; j++) cout << x[i][j] << " "; cout << endl; } system("pause"); return 0; } </pre>
Go	
<pre> package main import "fmt" func main() { var n,m int x := [10][10]int{} </pre>	


```

fmt.Print("m, n(<10)")
fmt.Scan(&m, &n)
for j:=0; j<n; j++ {
    fmt.Scan(&x[0][j])
}
for i:=1; i<m; i++ {
    for j:=0; j<n; j++ {
        x[i][j] = x[i-1][j]
    }
}
for i:=0; i<m; i++ {
    for j:=0; j<n; j++ {
        fmt.Printf("%+v ", x[i][j])
    }
    fmt.Println()
}
}

```

3) Дана квадратная матрица A порядка M . Начиная с элемента $A_{1,1}$, вывести ее элементы следующим образом («уголками»): все элементы первой строки; элементы последнего столбца, кроме первого (уже выведенного) элемента; оставшиеся элементы второй строки; оставшиеся элементы предпоследнего столбца и т. д.; последним выводится элемент $A_{M,1}$.

PascalABC.NET	C++
<pre> Var a: array [0..10, 0..10] of integer; M, i, j, i2: Integer; begin Write('M: '); Readln(M); M := M - 1; for i := 0 to M do begin for j := 0 to M do begin a[i, j] := Random(20); write(a[i, j], ' '); end; writeln; end; writeln; for i := 0 to M do begin for j := 0 to M - i do write(' ', a[i, j]); for var k := i + 1 to M do write(' ', a[k, M - i]); writeln(''); end; end. </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, m, i, j; cin >> m; m = m - 1; x = new int*[m+2]; for (i = 0; i <=m; i++) x[i] = new int[m+2]; for (i = 0; i <=m; i++) for (j = 0; j <= m; j++) cin>>x[i][j]; for (i = 0; i <= m; i++) { for (j = 0; j <=m; j++) cout << x[i][j] << " "; cout << endl; } cout << endl; for (i = 0; i <= m; i++) { for (j = 0; j <=m-i; j++) cout << " "<<x[i][j]; for (int k=i+1; k<=m; k++) cout << " " << x[k][m-i]; cout << endl; } system("pause"); return 0; } </pre>

```
Go
```

```

package main
import (
    "fmt"
    "math/rand"
)
func main() {
    var m,i,j,k int
    x := [10][10]int{}
    fmt.Print("m(<10)")
    fmt.Scan(&m)
    for i=0; i<m; i++ {
        for j=0; j<m; j++ {
            x[i][j] = rand.Intn(20)
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
    fmt.Println()
    for i=0; i<m; i++ {
        for j=0; j<m-i; j++ {
            fmt.Printf("%v ", x[i][j])
        }
        for k=i+1; k<m; k++ {
            fmt.Printf("%v ", x[k][m-1-i])
        }
        fmt.Println()
    }
}

```

4) Дана квадратная матрица A порядка M (M — нечетное число). Начиная с элемента $A_{1,1}$ и перемещаясь по часовой стрелке, вывести все ее элементы *по спирали*: первая строка, последний столбец, последняя строка в обратном порядке, первый столбец в обратном порядке, оставшиеся элементы второй строки и т. д.; последним выводится центральный элемент матрицы.

PascalABC.NET	C++
<pre> Var a: array [0..10, 0..10] of integer; M, i, j, k: Integer; begin Write('M: '); Readln(M); M := M - 1; for i := 0 to M do begin for j := 0 to M do begin a[i,j]:=Random(20); write(a[i, j], ' '); end; end; </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, m, i, j, k; cin >> m; m = m - 1; x = new int*[m+2]; for (i = 0; i <=m; i++) x[i] = new int[m+2]; for (i = 0; i <=m; i++) for (j = 0; j <= m; j++) x[i][j]=rand()%50; for (i = 0; i <= m; i++) { for (j = 0; j <=m; j++) cout << x[i][j] << " "; } } </pre>

<pre>writeln; end; writeln; for i := 0 to M div 2 do begin for j:=i to M-i do Write(a[i, j], ' '); for k:=i+1 to M-i do Write(a[k,M-i], ' '); for j:=M-1-i downto i do Write(a[M-i,j], ' '); for k:=M-1-i downto i+1 do Write(a[k, i], ' '); writeln(''); end; end.</pre>	<pre> cout << endl; } cout << endl; for (i = 0; i <= m/2; i++) { for (j = i; j <= m-i; j++) cout <<x[i][j]<<" "; for (k=i+1; k<=m-i; k++) cout << x[k][m-i]<<" "; for (j=m-1-i; j>=i; j--) cout << x[m-i][j] << " "; for (k=m-1-i; k>=i+1; k--) cout << x[k][i] << " "; cout << endl; } system("pause"); return 0; } }</pre>
Go	
<pre>package main import ("fmt" "math/rand") func main() { var m,i,j,k int x := [10][10]int{} fmt.Scan(&m) for i=0; i<m; i++ { for j=0; j<m; j++ { x[i][j] = rand.Intn(20) fmt.Printf("%v ", x[i][j]) } fmt.Println() } fmt.Println() for i=0; i<m/2; i++ { for j=0; j<m-i; j++ { fmt.Printf("%v ", x[i][j]) } for k=i+1; k<m-i; k++ { fmt.Printf("%v ", x[k][m-1-i]) } for j=m-2-i; j >=i; j--{ fmt.Printf("%v ", x[m-1-i][j]) } for k=m-2-i; k>=i+1; k-- { fmt.Printf("%v ", x[k][i]) } fmt.Println() } }</pre>	

2. Анализ элементов матрицы

5) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq M$). Найти сумму и произведение элементов K -й строки данной матрицы.

PascalABC.NET	C++
<pre> Var a: array [1..10, 1..10] of integer; M, N, K, i, j, i2: Integer; begin Write('M,N,K: '); Readln(M, N, K); for i := 1 to M do begin for j := 1 to N do begin a[i,j]:=Random(20); write(a[i, j], ' '); end; writeln; end; writeln; var S := 0; var P := 1; for j := 1 to N do begin S := S + a[K, j]; P := P * a[K, j]; end; Write('S=', S, 'P=', P); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, m, n,k, i, j, S=0, P=1; cin >> m>>n>>k; x = new int*[m]; for (i = 0; i <m; i++) x[i] = new int[n]; for (i = 0; i <m; i++) for (j = 0; j <n; j++) x[i][j]=rand()%50; for (i = 0; i <m; i++) { for (j = 0; j <n; j++) cout << x[i][j] << " "; cout << endl; } cout << endl; for (j = 0; j < n; j++) { S = S + x[k-1][j]; P = P*x[k-1][j]; } cout << S << " " << P << endl; system("pause"); return 0; } </pre>
C#	
<pre> static void Main(){ int m, n, k, i, j, sum = 0, mult = 1; Console.Write("M="); m = Convert.ToInt32(Console.ReadLine()); Console.Write("N="); n = Convert.ToInt32(Console.ReadLine()); Console.Write("K="); k = Convert.ToInt32(Console.ReadLine()); // Объявляем массив int[,] A = new int[m, n]; // Вводим массив for (i = 0; i < m; i++) { for (j = 0; j < n; j++) { Console.Write("A[" + (i + 1) + "," + (j + 1) + "]="); A[i, j] = Convert.ToInt32(Console.ReadLine()); } } } </pre>	

```

// Считаем сумму и произведение элементов k-й строки
for (j = 0; j < n; j++)
{
    sum += A[k - 1, j];
    mult *= A[k - 1, j];
}
Console.WriteLine("Сумма " + sum);
Console.WriteLine("Произведение " + mult);
}
}
}

```

Go

```

package main
import (
    "fmt"
    "math/rand"
)
func main() {
    var m,i,j,k int
    x := [10][10]int{}
    fmt.Print("m(<10)")
    fmt.Scan(&m)
    for i=0; i<m; i++ {
        for j=0; j<m; j++ {
            x[i][j] = rand.Intn(20)
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
    fmt.Println()
    fmt.Print("k(<10)")
    fmt.Scan(&k)
    var S,P int=0,1
    for j=0;j<m ; j++){
        S+=x[k-1][j];
        P*=x[k-1][j];
    }
    fmt.Printf("%d %d", S,P)
}

```

- 6) Дана целочисленная матрица размера $M \times N$, элементы которой могут принимать значения от 0 до 100. Различные строки матрицы назовем *похожими*, если совпадают множества чисел, встречающихся в этих строках. Найти количество строк, похожих на первую строку данной матрицы.

PascalABC.NET

```

Var
    a: array [1..10, 1..10] of integer;
    Num, M, N, i, j: Integer;
    L1, LM: set of byte;
begin
    Write('N,M: ');

```

```

Readln(N, M);
for i := 1 to M do
  for j := 1 to N do
    Read(a[i, j]);
writeln;
for i := 1 to M do
begin
  for j := 1 to N do
  begin
    write(a[i, j], ' ');
  end;
  writeln;
end;
writeln;
L1 := [];
for j := 1 to N do
  L1 := L1 + [a[1, j]];
Num := 0;
for i := 2 to M do
begin
  LM := [];
  for j := 1 to N do
    LM := LM + [a[i, j]];
  if L1 = LM then
    inc(Num);
end;
Writeln(Num);
end.

```

C++

```

#include <iostream>
using namespace std;
void sort(int a[], int n) {
  for (int i = 1; i < n; ++i)
    for (int j = 0; j < n - i; j++)
      if (a[j] > a[j + 1]){
        int temp = a[j];
        a[j] = a[j + 1];
        a[j + 1] = temp;
      }
}
bool isLike(int a[], int b[], int n) {
  sort(a, n);
  sort(b, n);
  int i = 0;
  while ((a[i] == b[i]) && (i < n))
    i++;
  return i == n;
}
int main() {
  int **a, m, n, i, j, k=0;

```

```

cin >> m>>n;
a = new int*[m];
for (i = 0; i < m; i++)
    a[i] = new int[n];
for (i = 0; i < m; ++i)
    for (j=0; j<n; j++)
        cin>>a[i][j];
cout << endl;
for (i = 0; i < m; ++i) {
    for (j = 0; j < n; j++)
        cout << a[i][j] << " ";
    cout << endl;
}
for (i = 0; i < m; i++)
    if (isLike(a[0], a[i], n))
        k++;
cout <<"k="<<k<<endl;
for (i = 0; i < m; i++)
    delete a[i];
delete a;
system("pause");
return 0;
}

```

7) Дана целочисленная матрица размера $M \times N$. Найти количество ее строк, все элементы которых различны.

PascalABC.NET

```

Var
a: array [1..10, 1..10] of integer;
Rez, M, N, i, j: Integer;
Row: set of byte;

begin
Write('N,M: ');
Readln(N, M);
for i := 1 to M do
begin
    for j := 1 to N do
    begin
        a[i, j] := Random(20);
        write(a[i, j], ' ');
    end;
    writeln;
end;
writeln;

Rez := 0;
for i := 1 to M do
begin
    j := 1;

```

```

Row := [];
while (not (a[i, j] in Row)) and (j <= N) do
begin
  Row := Row + [a[i, j]];
  inc(j);
end;
if J = N + 1 then
  inc(Rez);
end;
Writeln(Rez);
end.

```

C++

```

#include <iostream>
using namespace std;
void sort(int a[], int n) {
  for (int i = 1; i < n; ++i)
    for (int j = 0; j < n - i; j++)
      if (a[j] > a[j + 1]){
        int temp = a[j];
        a[j] = a[j + 1];
        a[j + 1] = temp;
      }
}
bool isUnique(int a[], int n) {
  sort(a, n);
  int i = 1;
  while ((a[i-1]<a[i]) && (i<n))
    i++;
  return i == n;
}
int main() {
  int **a, m,n, i,j, k=0;
  cin >> m>>n;
  a = new int*[m];
  for (i = 0; i < m; i++)
    a[i] = new int[n];
  for (i = 0; i < m; ++i)
    for (j=0; j<n; j++)
      cin>>a[i][j];
  cout << endl;
  for (i = 0; i < m; ++i) {
    for (j = 0; j < n; j++)
      cout << a[i][j] << " ";
    cout << endl;
  }
  for (i = 0; i < m; i++)
    if (isUnique(a[i], n))
      k++;
  cout <<"k="<<k<<endl;
  for (i = 0; i < m; i++)

```



```

        delete a[i];
    delete a;
    system("pause");
    return 0;
}

```

3. Преобразование матрицы

При выполнении заданий из данного пункта не следует использовать вспомогательные двумерные массивы-матрицы.

8) Дана матрица размера $M \times N$ и целые числа K_1 и K_2 ($1 \leq K_1 < K_2 \leq M$). Поменять местами строки матрицы с номерами K_1 и K_2 .

PascalABC.NET	C++
<pre> Type matrix = array [1..10, 1..10] of integer; var a: matrix; M,N,i,j,K1,K2: Integer; begin Write('N,M: '); Readln(N, M); for i := 1 to M do begin for j := 1 to N do begin a[i,j]:=Random(20); write(a[i, j], ' '); end; writeln; end; writeln; writeln('K1,K2'); Read(K1, K2); for j := 1 to N do swap(a[K1,j], a[K2,j]); for i := 1 to M do begin for j := 1 to N do begin Write(a[i, j], ' '); end; writeln(); end; end. </pre>	<pre> #include <iostream> using namespace std; void swap(int &a, int &b) { int h = a; a = b; b = h; } int main() { int **x, m, n, i, j, k1, k2; cin >> m >> n; x = new int*[m]; for (i = 0; i < m; i++) x[i] = new int[n]; for (i = 0; i < m; i++) for (j = 0; j < n; j++) x[i][j]=rand() % 50; for (i = 0; i < m; i++) { for (j = 0; j < n; j++) cout << x[i][j] << " "; cout << endl; } cout << "k1,k2: \n"; cin >> k1 >> k2; for (j = 0; j < n; j++) swap(x[k1][j], x[k2][j]); cout << endl; for (i = 0; i < m; i++) { for (j = 0; j < n; j++) cout << x[i][j] << " "; cout << endl; } system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math/rand") func main() { </pre>	

```

var m,i,j, k1, k2 int
x := [10][10]int{}
fmt.Print("m(<10)")
fmt.Scan(&m)
for i=0; i<m; i++ {
  for j=0; j<m; j++ {
    x[i][j] = rand.Intn(20)
    fmt.Printf("%+v ", x[i][j])
  }
  fmt.Println()
}
fmt.Println()
fmt.Print("k1, k2(<10)")
fmt.Scan(&k1, &k2)
for j:=0; j<m; j++){
  x[k1][j], x[k2][j]=x[k2][j], x[k1][j]
}
for i=0; i<m; i++ {
  for j=0; j<m; j++ {
    fmt.Printf("%+v ", x[i][j])
  }
  fmt.Println()
}
}
}

```

9) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq M$). Удалить строку матрицы с номером K .

PascalABC.NET	C++
<pre> Var a: array [1..10, 1..10] of integer; K, M, N, i, j: Integer; begin Write('N,M,K: '); Readln(N, M, K); for i := 1 to M do begin for j := 1 to N do begin a[i,j]:=Random(20); write(a[i,j], ''); end; writeln; end; writeln; M := M - 1; for i := K to M do for j := 1 to N do a[i,j]:=a[i+1,j]; for i := 1 to M do begin </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, m, n, i, j, k; cin >> m >> n; x = new int*[m]; for (i = 0; i < m; i++) x[i] = new int[n]; for (i = 0; i < m; i++) for (j = 0; j < n; j++) x[i][j] = rand() % 50; for (i = 0; i < m; i++) { for (j = 0; j < n; j++) cout << x[i][j] << " "; cout << endl; } cout << "k="; cin >> k; m = m - 1; for (i = k - 1; i < m; i++) for (j = 0; j < n; j++) x[i][j] = x[i + 1][j]; cout << endl; for (i = 0; i < m; i++) { for (j = 0; j < n; j++) </pre>

<pre> for j := 1 to N do begin Write(a[i,j], ''); end; writeln(); end; end. </pre>	<pre> cout << x[i][j] << " "; cout << endl; } system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math/rand") func main() { var m,n,i,j, k int x := [10][10]int{} fmt.Print("m, n(<10)") fmt.Scan(&m, &n) for i=0; i<m; i++ { for j=0; j<n; j++ { x[i][j] = rand.Intn(20) fmt.Printf("%+v ", x[i][j]) } fmt.Println() } fmt.Println() fmt.Print("k(<10)") fmt.Scan(&k) m=m-1 for i=k; i<m; i++ { for j:=0; j<n; j++){ x[i][j]=x[i+1][j] } } for i=0; i<m; i++ { for j=0; j<n; j++ { fmt.Printf("%+v ", x[i][j]) } fmt.Println() } } </pre>	

10) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq M$). Перед строкой матрицы с номером K вставить строку из нулей.

PascalABC.NET
<pre> Var a: array [1..10, 1..10] of integer; K, M, N, i, j: Integer; begin Write('N,M,K: '); Readln(N, M, K); </pre>

```

for i := 1 to M do
begin
  for j := 1 to N do
  begin
    a[i, j] := Random(20);
    write(a[i, j], ' ');
  end;
  writeln;
end;
writeln;
if k > 0 then
begin
  M := M + 1;
  for i := M downto k + 1 do
    for j := 1 to N do
      a[i, j] := a[i - 1, j];
    end;
  for j := 1 to N do
    a[k, j] := 0;
  for i := 1 to M do
  begin
    for j := 1 to N do
    begin
      Write(a[i, j], ' ');
    end;
    writeln();
  end;
end.

```

C++

```

#include <iostream>
using namespace std;
int main() {
  int **x, m, n, i, j, k;
  cin >> m >> n;
  x = new int*[m+2];
  for (i = 0; i < m+2; i++)
    x[i] = new int[n];
  for (i = 0; i < m; i++)
    for (j = 0; j < n; j++)
      x[i][j] = rand() % 50;
  for (i = 0; i < m; i++) {
    for (j = 0; j < n; j++)
      cout << x[i][j] << " ";
    cout << endl;
  }
  cout << "k="; cin >> k;
  k = k - 1;
  m = m + 1;
  for (i = m - 1; i > k; i--)
    for (j = 0; j < n; j++)
      x[i][j] = x[i - 1][j];
  for (j = 0; j < n; j++)
    x[k][j] = 0;
  cout << endl;
}

```

```

    for (i = 0; i < m; i++) {
        for (j = 0; j < n; j++)
            cout << x[i][j] << " ";
        cout << endl;
    }
    system("pause");
    return 0;
}

```

Go

```

package main
import (
    "fmt"
    "math/rand"
)
func main() {
    var m,n,i,j, k int
    x := [10][10]int{}
    fmt.Print("m, n(<10)")
    fmt.Scan(&m, &n)
    for i=0; i<m; i++ {
        for j=0; j<n; j++ {
            x[i][j] = rand.Intn(20)
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
    fmt.Println()
    fmt.Print("k(<10)")
    fmt.Scan(&k)
    m=m+1
    for i=m-1; i>k; i--{
        for j:=0; j<n; j++){
            x[i][j]=x[i-1][j]
        }
    }
    for j=0; j<n; j++){
        x[k][j]=0
    }
    for i=0; i<m; i++ {
        for j=0; j<n; j++ {
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
}

```

- 11) Дана матрица размера $M \times N$. Элемент матрицы называется ее *локальным минимумом*, если он меньше всех окружающих его элементов. Заменить все локальные минимумы данной матрицы на нули. При решении допускается использовать вспомогательную матрицу.

```

Type
  matrix = array [1..10, 1..10] of integer;
var
  a: matrix;
  K, M, N, i, j: Integer;
function LocMinMatr(var mat: matrix; X, Y: Integer): Boolean;
begin
  LocMinMatr := true;
  if x > 1 then
    if mat[x, y] >= mat[x - 1, y] then
      LocMinMatr := False;
  if x = mat[x + 1, y] then
    LocMinMatr := False;
  if y > 1 then
    if mat[x, y] >= mat[x, y - 1] then
      LocMinMatr := False;
  if y = mat[x, y + 1] then
    LocMinMatr := False;
  if (y > 1) and (x > 1) then
    if mat[x, y] >= mat[x - 1, y - 1] then
      LocMinMatr := False;
  if (y < N) and (x < M) then
    if mat[x, y] >= mat[x + 1, y + 1] then
      LocMinMatr := False;
  if (y > 1) and (x < M) then
    if mat[x, y] >= mat[x + 1, y - 1] then
      LocMinMatr := False;
  if (y < N) and (x > 1) then
    if mat[x, y] >= mat[x - 1, y + 1] then
      LocMinMatr := False;
end;

begin
  Write('N,M: ');
  Readln(N, M);
  for i := 1 to M do
    begin
      for j := 1 to N do
        begin
          a[i, j] := Random(20);
          write(a[i, j], ' ');
        end;
        writeln;
      end;
    end;
    writeln;

  for i := 1 to M do
    for j := 1 to N do
      if LocMinMatr(a, i, j) then

```

```

        a[i, j] := 0;

    for i := 1 to M do
    begin
        for j := 1 to N do
        begin
            Write(a[i, j], ' ');
        end;
        writeln;
    end;
end.

```

C++

```

#include <iostream>
using namespace std;
int main() {
    int **x, m, n, i, j, k;
    cin >> m >> n;
    x = new int*[m+2];
    for (i = 0; i < m+2; i++)
        x[i] = new int[n];
    for (i = 0; i < m; i++)
        for (j = 0; j < n; j++)
            x[i][j] = rand() % 50;
    for (i = 0; i < m; i++) {
        for (j = 0; j < n; j++)
            cout << x[i][j] << " ";
        cout << endl;
    }
    int num;
    for (i = 0; i < m; ++i) {
        for (j = 0; j < n; ++j) {
            num = 0;
            if (i - 1 >= 0) {
                if (x[i][j] < x[i - 1][j]) num++;
            }
            else ++num;
            if ((i + 1) < m) {
                if (x[i][j] < x[i + 1][j]) num++;
            }
            else ++num;
            if (j - 1 >= 0) {
                if (x[i][j] < x[i][j - 1]) num++;
            }
            else ++num;
            if ((j + 1) < n) {
                if (x[i][j] < x[i][j + 1]) num++;
            }
            else
                ++num;
            if (num == 4) x[i][j] = 0;
        }
    }
}

```

```

    }
}
cout << endl;
for (i = 0; i < m; i++) {
    for (j = 0; j < n; j++)
        cout << x[i][j] << " ";
    cout << endl;
}
for (i = 0; i < m + 2; i++)
    delete x[i];
delete x;
system("pause");
return 0;
}

```

4. Диагонали квадратной матрицы

12) Дана квадратная матрица A порядка N . Найти сумму элементов ее *главной диагонали*, то есть диагонали, содержащей следующие элементы:

$$A_{1,1}, A_{2,2}, A_{3,3}, \dots, A_{M,M}.$$

PascalABC.NET	C++
<pre> Var a: array [1..10, 1..10] of integer; N,i,j: Integer; begin Write('N '); Readln(N); for i := 1 to N do begin for j := 1 to N do begin a[i, j] := Random(20); write(a[i, j], ' '); end; writeln; end; writeln; var S:=0; for i:=1 to N do S+=a[i,i]; WriteFormat('Sum={0}',S); end. </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, m, i, j, S = 0; cin >> m; x = new int*[m]; for (i = 0; i < m; i++) x[i] = new int[m]; for (i = 0; i < m; i++) for (j = 0; j < m; j++) x[i][j] = rand() % 50; for (i = 0; i < m; i++) { for (j = 0; j < m; j++) cout << x[i][j] << " "; cout << endl; } for (i = 0; i < m; i++) S += x[i][i]; cout << S << endl; for (i = 0; i < m; i++) delete x[i]; delete x; system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math/rand") </pre>	


```

func main() {
    var m,i,j int
    x := [10][10]int{}
    fmt.Print("m, n(<10)")
    fmt.Scan(&m)
    for i=0; i<m; i++ {
        for j=0; j<m; j++ {
            x[i][j] = rand.Intn(20)
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
    fmt.Println()
    S:=0
    for j=0; j<m; j++){
        S+=x[j][j]
    }
    fmt.Println(S)
}

```

- 13) Дана квадратная матрица A порядка N . Найти сумму элементов каждой ее диагонали, параллельной главной (начиная с одноэлементной диагонали $A_{1,M}$).

PascalABC.NET	C++
<pre> Var a:array [1..10,1..10] of integer; S,N, i, j:Integer; Begin Write('N '); Readln(N); for i := 1 to N do begin for j := 1 to N do begin a[i,j]:=Random(20); write(a[i, j], ' '); end; writeln; end; writeln; for j:=1 to N-1 do begin S:=0; for i:=1 to j do begin S:=S+a[i,N-(j-i)]; end; Writeln(S); end; for j:=N-1 downto 1 do </pre>	<pre> #include <iostream> using namespace std; int main() { int **x, m, i, j, S = 0; cin >> m; x = new int*[m]; for (i = 0; i < m; i++) x[i] = new int[m]; for (i = 0; i < m; i++) for (j = 0; j < m; j++) x[i][j] = rand() % 50; for (i = 0; i < m; i++) { for (j = 0; j < m; j++) cout << x[i][j] << " "; cout << endl; } m--; for (j = 0; j <= m; j++) { S = 0; for (i = 0; i <= j; i++) S += x[i][m - (j - i)]; cout << "S=" << S << endl; } for (j = m - 1; j >= 0; j--) { S = 0; for (i = 0; i <= j; i++) S += x[m - (j - i)][i]; cout << "S=" << S << endl; } for (i = 0; i < m; i++) </pre>

<pre> begin S:=0; for i:=1 to j do begin S:=S+a[N-(j-i),i]; end; writeln(S); end; end. </pre>	<pre> delete x[i]; delete x; system("pause"); return 0; } </pre>
Go	
<pre> package main import ("fmt" "math/rand") func main() { var m,i,j int x := [10][10]int{} fmt.Print("m(<10)") fmt.Scan(&m) for i=0; i<m; i++ { for j=0; j<m; j++ { x[i][j] = rand.Intn(20) fmt.Printf("%+v ", x[i][j]) } fmt.Println() } fmt.Println() m-- for j=0; j<=m; j++){ S:=0 for i=0; i<=j; i++){ S+=x[i][m-(j-i)] } fmt.Println(S) } for j=m-1; j>=0; j--{ S:=0 for i=0; i<=j; i++){ S+=x[m-(j-i)][i] } fmt.Println(S) } } </pre>	

14) Дана квадратная матрица A порядка M . Зеркально отразить ее элементы относительно побочной диагонали. (при этом элементы побочной диагонали останутся на прежнем месте, элемент $A_{1,1}$ поменяется местами с $A_{M,M}$, элемент $A_{1,2}$ — с $A_{M-1,M}$ и т. д.). Вспомогательную матрицу не использовать.

PascalABC.NET

```

Type
  matrix = array [1..10, 1..10] of integer;
var
  a: matrix;
  M, i, j: Integer;
begin
  Write('M: ');
  Readln(M);
  for i := 1 to M do
  begin
    for j := 1 to M do
    begin
      a[i, j] := Random(20);
      write(a[i, j], ' ');
    end;
    writeln;
  end;
  writeln;
  for j := 1 to M do
    for i := 1 to M - j do
      Swap(a[i, j], a[M - j + 1, M - i + 1]);
  for i := 1 to M do
  begin
    for j := 1 to M do
    begin
      Write(a[i,j], ' ');
    end;
    writeln();
  end;
end.

```

C++

```

#include <iostream>
using namespace std;
void swap(int &a, int &b) {
    int h = a;
    a = b;
    b = h;
}
int main() {
    int **x, m, i, j, S = 0;
    cin >> m;
    x = new int*[m];
    for (i = 0; i < m; i++)
        x[i] = new int[m];
    for (i = 0; i < m; i++)
        for (j = 0; j < m; j++)
            x[i][j] = rand() % 50;
    for (i = 0; i < m; i++) {
        for (j = 0; j < m; j++)
            cout << x[i][j] << " ";
        cout << endl;
    }
    m--;
}

```

```

    for (j = 0; j <= m; j++)
        for (i = 0; i <= m - j; i++)
            swap(x[i][j], x[m - j][m - i]);
    cout << endl;
    m++;
    for (i = 0; i < m; i++) {
        for (j = 0; j < m; j++)
            cout << x[i][j] << " ";
        cout << endl;
    }
    for (i = 0; i < m; i++)
        delete x[i];
    delete x;
    system("pause");
    return 0;
}

```

Go

```

package main
import (
    "fmt"
    "math/rand"
)
func main() {
    var m,i,j int
    x := [10][10]int{}
    fmt.Print("m(<10)")
    fmt.Scan(&m)
    for i=0; i<m; i++ {
        for j=0; j<m; j++ {
            x[i][j] = rand.Intn(20)
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
    fmt.Println()
    m--
    for j=0; j<=m; j++){
        for i=0; i<=m-j; i++){
            x[i][j],x[m-j][m-i]=x[m-j][m-i],x[i][j]
        }
    }
    for i=0; i<=m; i++ {
        for j=0; j<=m; j++ {
            fmt.Printf("%+v ", x[i][j])
        }
        fmt.Println()
    }
}

```

Задачи для самостоятельной работы

Формирование матрицы и вывод ее элементов

В заданиях на формирование матрицы предполагается, что размер результирующей матрицы не превосходит 10.

- 1) Даны целые положительные числа M и N . Сформировать целочисленную матрицу размера $M \times N$, у которой все элементы J -го столбца имеют значение $5 \cdot J$ ($J = 1, \dots, N$).
- 2) Даны целые положительные числа M , N и набор из N чисел. Сформировать матрицу размера $M \times N$, у которой в каждой строке содержатся все числа из исходного набора (в том же порядке).
- 3) Даны целые положительные числа M , N , число D и набор из M чисел. Сформировать матрицу размера $M \times N$, у которой первый столбец совпадает с исходным набором чисел, а элементы каждого следующего столбца равны сумме соответствующего элемента предыдущего столбца и числа D (в результате каждая строка матрицы будет содержать элементы *арифметической прогрессии*).
- 4) Даны целые положительные числа M , N , число Q и набор из N чисел. Сформировать матрицу размера $M \times N$, у которой первая строка совпадает с исходным набором чисел, а элементы каждой следующей строки равны соответствующему элементу предыдущей строки, умноженному на Q (в результате каждый столбец матрицы будет содержать элементы *геометрической прогрессии*).
- 5) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq M$). Вывести элементы K -й строки данной матрицы.
- 6) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq N$). Вывести элементы K -го столбца данной матрицы.
- 7) Дана матрица размера $M \times N$. Вывести ее элементы, расположенные в строках с четными номерами (2, 4, ...). Вывод элементов производить по строкам, условный оператор не использовать.
- 8) Дана матрица размера $M \times N$. Вывести ее элементы, расположенные в столбцах с нечетными номерами (1, 3, ...). Вывод элементов производить по столбцам, условный оператор не использовать.
- 9) Дана матрица размера $M \times N$. Вывести ее элементы в следующем порядке: первая строка слева направо, вторая строка справа налево, третья строка слева направо, четвертая строка справа налево и т. д.
- 10) Дана матрица размера $M \times N$. Вывести ее элементы в следующем порядке: первый столбец сверху вниз, второй столбец снизу вверх, третий столбец сверху вниз, четвертый столбец снизу вверх и т. д.
- 11) Дана квадратная матрица A порядка M . Начиная с элемента $A_{1,1}$, вывести ее элементы следующим образом («*уголками*»): все элементы первого столбца; элементы последней строки, кроме первого (уже выведенного) элемента; оставшиеся элементы второго столбца; оставшиеся элементы предпоследней строки и т. д.; последним выводится элемент $A_{1,M}$.
- 12) Дана квадратная матрица A порядка M (M — нечетное число). Начиная с элемента $A_{1,1}$ и перемещаясь против часовой стрелки, вывести все ее элементы *по спирали*: первый столбец, последняя строка, последний столбец в обратном порядке, первая строка в обратном порядке, оставшиеся элементы второго столбца и т. д.; последним выводится центральный элемент матрицы.

Анализ элементов матрицы

- 13) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq N$). Найти сумму и произведение элементов K -го столбца данной матрицы.
- 14) Дана матрица размера $M \times N$. Для каждой строки матрицы найти сумму ее элементов.
- 15) Дана матрица размера $M \times N$. Для каждого столбца матрицы найти произведение его элементов.
- 16) Дана матрица размера $M \times N$. Для каждой строки матрицы с нечетным номером (1, 3, ...) найти среднее арифметическое ее элементов. Условный оператор не использовать.
- 17) Дана матрица размера $M \times N$. Для каждого столбца матрицы с четным номером (2, 4, ...) найти сумму его элементов. Условный оператор не использовать.
- 18) Дана матрица размера $M \times N$. В каждой строке матрицы найти минимальный элемент.
- 19) Дана матрица размера $M \times N$. В каждом столбце матрицы найти максимальный элемент.
- 20) Дана матрица размера $M \times N$. Найти номер ее строки с наибольшей суммой элементов и вывести данный номер, а также значение наибольшей суммы.
- 21) Дана матрица размера $M \times N$. Найти номер ее столбца с наименьшим произведением элементов и вывести данный номер, а также значение наименьшего произведения.
- 22) Дана матрица размера $M \times N$. Найти максимальный среди минимальных элементов ее строк.
- 23) Дана матрица размера $M \times N$. Найти минимальный среди максимальных элементов ее столбцов.
- 24) Дана матрица размера $M \times N$. В каждой ее строке найти количество элементов, меньших среднего арифметического всех элементов этой строки.
- 25) Дана матрица размера $M \times N$. В каждом ее столбце найти количество элементов, больших среднего арифметического всех элементов этого столбца.
- 26) Дана матрица размера $M \times N$. Найти номера строки и столбца для элемента матрицы, наиболее близкого к среднему значению всех ее элементов.
- 27) Дана целочисленная матрица размера $M \times N$. Найти номер первой из ее строк, содержащих равное количество положительных и отрицательных элементов (нулевые элементы матрицы не учитываются). Если таких строк нет, то вывести 0.
- 28) Дана целочисленная матрица размера $M \times N$. Найти номер последнего из ее столбцов, содержащих равное количество положительных и отрицательных элементов (нулевые элементы матрицы не учитываются). Если таких столбцов нет, то вывести 0.
- 29) Дана целочисленная матрица размера $M \times N$. Найти номер последней из ее строк, содержащих только четные числа. Если таких строк нет, то вывести 0.
- 30) Дана целочисленная матрица размера $M \times N$. Найти номер первого из ее столбцов, содержащих только нечетные числа. Если таких столбцов нет, то вывести 0.
- 31) Дана целочисленная матрица размера $M \times N$, элементы которой могут принимать значения от 0 до 100. Различные столбцы матрицы назовем *похожими*,

- если совпадают множества чисел, встречающихся в этих столбцах. Найти количество столбцов, похожих на последний столбец данной матрицы.
- 32) Дана целочисленная матрица размера $M \times N$. Найти количество ее столбцов, все элементы которых различны.
 - 33) Дана целочисленная матрица размера $M \times N$. Найти номер последней из ее строк, содержащих максимальное количество одинаковых элементов.
 - 34) Дана целочисленная матрица размера $M \times N$. Найти номер первого из ее столбцов, содержащих максимальное количество одинаковых элементов.
 - 35) Дана матрица размера $M \times N$. Найти количество ее строк, элементы которых упорядочены по возрастанию.
 - 36) Дана матрица размера $M \times N$. Найти количество ее столбцов, элементы которых упорядочены по убыванию.
 - 37) Дана матрица размера $M \times N$. Найти минимальный среди элементов тех строк, которые упорядочены либо по возрастанию, либо по убыванию. Если упорядоченные строки в матрице отсутствуют, то вывести 0.
 - 38) Дана матрица размера $M \times N$. Найти максимальный среди элементов тех столбцов, которые упорядочены либо по возрастанию, либо по убыванию. Если упорядоченные столбцы в матрице отсутствуют, то вывести 0.
 - 39) Дана целочисленная матрица размера $M \times N$. Найти элемент, являющийся максимальным в своей строке и минимальным в своем столбце. Если такой элемент отсутствует, то вывести 0.

Преобразование матрицы

При выполнении заданий из данного пункта (за исключением заданий не следует использовать вспомогательные двумерные массивы-матрицы.

- 40) Дана матрица размера $M \times N$ и целые числа K_1 и K_2 ($1 \leq K_1 < K_2 \leq N$). Поменять местами столбцы матрицы с номерами K_1 и K_2 .
- 41) Дана матрица размера $M \times N$. Преобразовать матрицу, поменяв местами минимальный и максимальный элемент в каждой строке.
- 42) Дана матрица размера $M \times N$. Преобразовать матрицу, поменяв местами минимальный и максимальный элемент в каждом столбце.
- 43) Дана матрица размера $M \times N$. Поменять местами строки, содержащие минимальный и максимальный элементы матрицы.
- 44) Дана матрица размера $M \times N$. Поменять местами столбцы, содержащие минимальный и максимальный элементы матрицы.
- 45) Дана матрица размера $M \times N$. Поменять местами столбец с номером 1 и последний из столбцов, содержащих только положительные элементы. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 46) Дана матрица размера $M \times N$. Поменять местами столбец с номером N и первый из столбцов, содержащих только отрицательные элементы. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 47) Дана матрица размера $M \times N$ (M — четное число). Поменять местами верхнюю и нижнюю половины матрицы.
- 48) Дана матрица размера $M \times N$ (N — четное число). Поменять местами левую и правую половины матрицы.
- 49) Дана матрица размера $M \times N$ (M и N — четные числа). Поменять местами левую верхнюю и правую нижнюю четверти матрицы.

- 50) Дана матрица размера $M \times N$ (M и N — четные числа). Поменять местами левую нижнюю и правую верхнюю четверти матрицы.
- 51) Дана матрица размера $M \times N$. Зеркально отразить ее элементы относительно горизонтальной оси симметрии матрицы (при этом меняются местами строки с номерами 1 и M , 2 и $M - 1$ и т. д.).
- 52) Дана матрица размера $M \times N$. Зеркально отразить ее элементы относительно вертикальной оси симметрии матрицы (при этом меняются местами столбцы с номерами 1 и N , 2 и $N - 1$ и т. д.).
- 53) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq N$). Удалить столбец матрицы с номером K .
- 54) Дана матрица размера $M \times N$. Удалить строку, содержащую минимальный элемент матрицы.
- 55) Дана матрица размера $M \times N$. Удалить столбец, содержащий максимальный элемент матрицы.
- 56) Дана матрица размера $M \times N$. Удалить ее первый столбец, содержащий только положительные элементы. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 57) Дана матрица размера $M \times N$. Удалить ее последний столбец, содержащий только отрицательные элементы. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 58) Дана матрица размера $M \times N$, содержащая как положительные, так и отрицательные элементы. Удалить все ее столбцы, содержащие только положительные элементы. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 59) Дана матрица размера $M \times N$ и целое число K ($1 \leq K \leq N$). После столбца матрицы с номером K вставить столбец из единиц.
- 60) Дана матрица размера $M \times N$. Продублировать строку матрицы, содержащую ее максимальный элемент.
- 61) Дана матрица размера $M \times N$. Продублировать столбец матрицы, содержащий ее минимальный элемент.
- 62) Дана матрица размера $M \times N$. Перед первым столбцом, содержащим только положительные элементы, вставить столбец из единиц. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 63) Дана матрица размера $M \times N$. После последнего столбца, содержащего только отрицательные элементы, вставить столбец из нулей. Если требуемых столбцов нет, то вывести матрицу без изменений.
- 64) Дана матрица размера $M \times N$. Элемент матрицы называется ее *локальным максимумом*, если он больше всех окружающих его элементов. Поменять знак всех локальных максимумов данной матрицы на противоположный. При решении допускается использовать вспомогательную матрицу.
- 65) Дана матрица размера $M \times N$. Упорядочить ее строки так, чтобы их первые элементы образовывали возрастающую последовательность.
- 66) Дана матрица размера $M \times N$. Упорядочить ее столбцы так, чтобы их последние элементы образовывали убывающую последовательность.
- 67) Дана матрица размера $M \times N$. Упорядочить ее строки так, чтобы их минимальные элементы образовывали убывающую последовательность.

68) Дана матрица размера $M \times N$. Упорядочить ее столбцы так, чтобы их максимальные элементы образовывали возрастающую последовательность.

Диагонали квадратной матрицы

69) Дана квадратная матрица A порядка M . Найти среднее арифметическое элементов ее *побочной диагонали*, то есть диагонали, содержащей следующие элементы:

$$A_{1,M}, A_{2,M-1}, A_{3,M-2}, \dots, A_{M,1}.$$

70) Дана квадратная матрица A порядка M . Найти сумму элементов каждой ее диагонали, параллельной *побочной* (начиная с одноэлементной диагонали $A_{1,1}$).

71) Дана квадратная матрица A порядка M . Найти среднее арифметическое элементов каждой ее диагонали, параллельной *главной* (начиная с одноэлементной диагонали $A_{1,M}$).

72) Дана квадратная матрица A порядка M . Найти среднее арифметическое элементов каждой ее диагонали, параллельной *побочной* (начиная с одноэлементной диагонали $A_{1,1}$).

73) Дана квадратная матрица A порядка M . Найти минимальный элемент для каждой ее диагонали, параллельной *главной* (начиная с одноэлементной диагонали $A_{1,M}$).

74) Дана квадратная матрица A порядка M . Найти минимальный элемент для каждой ее диагонали, параллельной *главной* (начиная с одноэлементной диагонали $A_{1,M}$).

75) Дана квадратная матрица A порядка M . Найти максимальный элемент для каждой ее диагонали, параллельной *побочной* (начиная с одноэлементной диагонали $A_{1,1}$).

76) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие ниже *главной диагонали*. Условный оператор не использовать.

77) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие выше *побочной диагонали*. Условный оператор не использовать.

78) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие на *побочной диагонали* и ниже нее. Условный оператор не использовать.

79) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие на *главной диагонали* и выше нее. Условный оператор не использовать.

80) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие одновременно выше *главной диагонали* и выше *побочной диагонали*. Условный оператор не использовать.

81) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие одновременно выше *главной диагонали* и ниже *побочной диагонали*. Условный оператор не использовать.

82) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие одновременно ниже *главной диагонали* (включая эту диагональ) и выше *побочной диагонали* (также включая эту диагональ). Условный оператор не использовать.

83) Дана квадратная матрица порядка M . Обнулить элементы матрицы, лежащие одновременно ниже *главной диагонали* (включая эту диагональ) и ниже *побочной диагонали* (также включая эту диагональ). Условный оператор не использовать.

- 84) Дана квадратная матрица A порядка M . Зеркально отразить ее элементы относительно главной диагонали (при этом элементы главной диагонали останутся на прежнем месте, элемент $A_{1,2}$ поменяется местами с $A_{2,1}$, элемент $A_{1,3}$ — с $A_{3,1}$ и т. д.). Вспомогательную матрицу не использовать.
- 85) Дана квадратная матрица A порядка M . Повернуть ее на угол 180° (при этом элемент $A_{1,1}$ поменяется местами с $A_{M,M}$, элемент $A_{1,2}$ — с $A_{M,M-1}$ и т. д.). Вспомогательную матрицу не использовать.
- 86) Дана квадратная матрица A порядка M . Повернуть ее на угол 90° в положительном направлении, то есть против часовой стрелки (при этом элемент $A_{1,1}$ перейдет в $A_{M,1}$, элемент $A_{M,1}$ — в $A_{M,M}$ и т. д.). Вспомогательную матрицу не использовать.
- 87) Дана квадратная матрица A порядка M . Повернуть ее на угол 90° в отрицательном направлении, то есть по часовой стрелке (при этом элемент $A_{1,1}$ перейдет в $A_{1,M}$, элемент $A_{1,M}$ — в $A_{M,M}$ и т. д.). Вспомогательную матрицу не использовать.

13. Символы и строки: группа String

При выполнении заданий на обработку русских букв можно считать, что буква «ё» в исходных строковых данных отсутствует. Это связано с тем, что расположение данного символа в кодовой таблице не соответствует его расположению в русском алфавите, и поэтому учет символа «ё» во многих алгоритмах требует дополнительных действий.

1. Символы и их коды. Формирование строк

1) Дан символ *C*. Вывести его *код* (то есть номер в кодовой таблице).

PascalABC.NET		C++	
<pre>var C: Char; begin Write('C:'); readln(C); Writeln('C:', ord(C)); end.</pre>		<pre>#include <iostream> #include <windows.h> using namespace std; int main() { SetConsoleCP(1251); SetConsoleOutputCP(1251); unsigned char C; int kod; cout << "Введите символ C: "; cin >> C; kod = C; cout << kod << endl; system("pause"); return 0; }</pre>	
Python	C#		
<pre>c=input() print(ord(c))</pre>	<pre>static void Main(string[] args) { char c = Convert.ToChar(Console.ReadLine()); int m = (int)c; Console.WriteLine(m); Console.ReadKey(); }</pre>		
Java		Go	
<pre>public static void main(String[] args) { Scanner sc=new Scanner(System.in); char c=sc.nextLine().charAt(0); int n=(int)c; System.out.println(n); }</pre>		<pre>package main import "fmt" func main() { var s string fmt.Scanln(&s) var n=int(s[0]) fmt.Println(n) }</pre>	
Kotlin			
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var c = reader.nextLine().get(0); var n = c.toInt() println(n) }</pre>			

2) Дано целое число N ($32 \leq N \leq 126$). Вывести символ с кодом, равным N .

PascalABC.NET		C++	
<pre>var N: byte; begin Write('N:'); readln(N); Writeln('N:', chr(N)); end.</pre>		<pre>#include <iostream> #include <windows.h> using namespace std; int main() { SetConsoleCP(1251); SetConsoleOutputCP(1251); unsigned char C; int n; cout << "Введите символ C: "; cin >> n; C=n; cout<<C<<endl; system("pause"); return 0; }</pre>	
Python	C#		
<pre>n=int(input()) print(chr(n))</pre>	<pre>static void Main(string[] args) { int m=Convert.ToInt32(Console.ReadLine()); int c = (char)m; Console.WriteLine(c); Console.ReadKey(); }</pre>		
Java		Go	
<pre>public static void main(String[] args) { Scanner sc=new Scanner(System.in); int n=sc.nextInt(); char c=(char)n; System.out.println(c); }</pre>		<pre>package main import "fmt" func main() { var n int fmt.Scanln(&n) ch := rune(n) fmt.Printf("char: %c\n", ch) }</pre>	
Kotlin			
<pre>import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var n = reader.nextInt() var c= n.toChar() println(c) }</pre>			
VB.NET			
<pre>Sub Main() Dim c As Char Dim n As Integer c = Convert.ToChar(Console.ReadLine()) n = Convert.ToInt32(c) Console.WriteLine(n) Console.ReadKey() End Sub</pre>			

3) Дано целое число N ($1 \leq N \leq 26$). Вывести N последних *строчных* (то есть маленьких) букв латинского алфавита в обратном порядке (начиная с буквы «Z»).

PascalABC.NET		C++	
<pre> Var N, i: byte; C: char; begin Write('N:'); readln(N); for i:=122 downto 122-N+1 do Writeln(chr(i)); end. </pre>		<pre> #include <iostream> using namespace std; int main() { int n, i; cin >> n; for (i = 122; i >= 122 - n; i--) cout << (char)i << " "; system("pause"); return 0; } </pre>	
Python		C#	
<pre> n=int(input()) for i in range (122,122- n+1,-1): print(chr(i)); </pre>		<pre> static void Main(string[] args) { int n; n=Convert.ToInt32(Console.ReadLine()); for (int i = 122; i >= 122 - n; i--) Console.Write((char)i+" "); Console.ReadKey(); } </pre>	
Java		Go	
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); int n=sc.nextInt(); for (int i=122; i>=122-n; i--){ char c=(char)i; System.out.print(c+" "); } } </pre>		<pre> package main import "fmt" func main() { var n int fmt.Scan(&n) for i:=122;i>=122-n+1;i--{ ch := rune(i) fmt.Printf("%c ", ch) } } </pre>	
VB.NET			
<pre> Sub Main() Dim c As Char Dim n As Integer n = Convert.ToInt32(Console.ReadLine()) c = Convert.ToChar(n) Console.WriteLine(c) Console.ReadKey() End Sub </pre>			

4) Дана непустая строка S . Вывести строку, содержащую символы строки S , между которыми вставлено по одному пробелу.

PascalABC.NET		C++	
<pre> var i: integer; S, S2: string; </pre>		<pre> #include "stdafx.h" #include <iostream> #include <string> using namespace std; </pre>	

<pre> begin Write('S:'); Readln(S); S2 := S[1]; for i := 2 to Length(S) do S2 := S2 + ' ' + S[i]; Writeln(S2); end. </pre>	<pre> int main(){ string s, Ss=""; getline(cin, s); for (int i = 0; i < s.length(); i++){ Ss = Ss+s[i] + " "; } cout << Ss; system("pause"); return 0; } </pre>
Python	C#
	<pre> static void Main(string[] args){ string s = Console.ReadLine(); int i = 1; while (i < s.Length){ s = s.Insert(i, " "); i += 2; } Console.WriteLine(s); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String S1=""; for (int i=0; i<S.length(); i++) S1=S1+S.charAt(i)+" "; System.out.print(S1); } </pre>	<pre> package main import ("bufio" "fmt" "os" "strings") func main() { reader := bufio.NewReader(os.Stdin) s, _:=reader.ReadString('\n') s=strings.Trim(s," ") var t string for i:= 0; i < len(s); i++ { t=t+string(s[i])+string("*"); } t=strings.Trim(t," *") fmt.Printf("%s",t) } </pre>
Kotlin	
<pre> import java.util.Scanner fun main(args: Array<String>) { val reader = Scanner(System.`in`) var S= reader.next() var S1="" for (i in (0..S.length-1)) S1=S1+S.get(i)+" "; println(S1) } </pre>	

2. Посимвольный анализ и преобразование строк.

Строки и числа

5) Дана строка. Подсчитать количество содержащихся в ней цифр.

PascalABC.NET		C++	
<pre> Var S: string; begin Write('S:'); Readln(S); var k := 0; for var i:=1 to Length(S) do if char.IsDigit(S[i]) then k += 1; Writeln(k); end. </pre>		<pre> #include <iostream> #include <string> using namespace std; int main(){ string s; getline(cin, s); int k = 0, i; for (i = 0; i < s.length(); i++) if (isdigit(s[i])) k++; cout<< k; system("pause"); return 0; } </pre>	
Python	C#		
<pre> S=input("S->") n=len(S) k=0 for i in range (0, n): if S[i].isdigit(): k+=1 print(k) </pre>	<pre> static void Main(string[] args){ string s = Console.ReadLine(); int k = 0; for (int i = 0; i < s.Length; i++) if (Char.IsDigit(s[i])) k++; Console.WriteLine(k); Console.ReadKey(); } </pre>		
Java		Go	
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); int i, k=0; for (i=0; i<S.length(); i++){ int m=(int)S.charAt(i); if ((m>47) && (m<58)) k++; } System.out.println(k); } </pre>		<pre> package main import ("bufio" "fmt" "os" "strings") func main() { reader:=bufio.NewReader(os.Stdin) s, _:=reader.ReadString('\n') s=strings.Trim(s, " ") var k int=0 for i:=0; i < len(s); i++ { if int(s[i])>=48 && int(s[i]) <=57{ k++ } } fmt.Println(k) } </pre>	

6) Дана строка. Подсчитать общее количество содержащихся в ней строчных латинских и русских букв.

PascalABC.NET	C++
<pre> Begin var S := readstring; var k := 0; for var i:=1 to Length(S)do if (ord(S[i]) in [97..122, 224..239]) then k += 1; Writeln(k); end. </pre>	<pre> #include <iostream> #include <string> #include <windows.h> using namespace std; bool lower(unsigned char c) { return (((int)c > 96) && ((int)c < 123)) (((int)c > 223) && ((int)c < 240)); } int main(){ SetConsoleCP(1251); SetConsoleOutputCP(1251); string s; unsigned char c; cout<<"ВВЕДИТЕ СИМВОЛ S: "; getline(cin, s); int k = 0, i; for (i = 0; i < s.length(); i++) { c = s[i]; if (lower(c)) k++; } cout << k; system("pause"); return 0; } </pre>
Python	C#
<pre> S=input("S->") n=len(S) k=0 for i in range (0, n): if S[i].isalpha(): k+=1 print(k) </pre>	<pre> static void Main(string[] args) { string s = Console.ReadLine(); int k = 0; for (int i = 0; i < s.Length; i++) if (Char.IsUpper(s[i])) if (s[i]<='Z') k++; Console.WriteLine(k); Console.ReadKey(); } </pre>
Go	
<pre> package main import ("bufio" "fmt" "os" "strings") func IsLetter(s string) int { var k int=0 for _, r := range s { if (r >= 'A' && r <= 'Z') (r >= 'a' && r <= 'z') (r >= 'A' && r <= 'Я') (r >= 'a' && r <= 'я'){ k++; } } return k } </pre>	


```

}
func main() {
    reader := bufio.NewReader(os.Stdin)
    s, _ := reader.ReadString('\n')
    s=strings.Trim(s," ")
    fmt.Println(IsLetter(s))
}

```

7) Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке слева направо).

PascalABC.NET		C++	
Var N: integer; S: string; begin Write('N:'); Readln(N); Str(N, S); Writeln(S); end.	#include <iostream> #include <string> using namespace std; int main(){ string s; getline(cin, s); int m = stoi(s); cout <<m; system("pause"); return 0; }		
Python		C#	
S=input("S->") n=int(S) print(n)	static void Main(string[] args){ string s = Console.ReadLine(); int m = Convert.ToInt32(s); Console.WriteLine(m); Console.ReadKey(); }		
Java		Go	
public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); int n=Integer.valueOf(S); System.out.println(n); }	package main import ("fmt" "strconv") func main() { var s string fmt.Scan(&s) n, _ := strconv.Atoi(s) fmt.Println(n) }		

8) Дана строка, изображающая арифметическое выражение вида «<цифра>±<цифра>±...±<цифра>», где на месте знака операции «±» находится символ «+» или «-» (например, «4+7-2-8»). Вывести значение данного выражения (целое число).

PascalABC.NET		C++	
Var S: String; Rez, op,i: integer; begin Read(S);	#include <iostream> #include <string> using namespace std; int main(){ string S; getline(cin, S);		

<pre> Rez := ord(S[1]) - 48; for i := 2 to Length(S) do if (char.IsDigit(S[i])) then Rez:=Rez+op*(ord(S[i])-48) else if (ord(S[i]) = 43) then op := 1 //+ else if (ord(S[i]) = 45) then op := -1; //- Writeln(Rez); end. </pre>	<pre> int Rez = int(S[0]) - 48; int i, op = 1; for (i = 1; i < S.size(); i++) { if (isdigit(S[i])) Rez = Rez+op*(int(S[i])-48); if (int(S[i]) == 43) op = 1; else if (int(S[i]) == 45) op = -1; } cout << Rez; system("pause"); return 0; } </pre>
Python	C#
<pre> S=input("S->") Rez=ord(S[0])-48 n=len(S) op=1 for i in range (1, n): if S[i].isdigit(): Rez=Rez+op* (ord(S[i])-48) elif ord(S[i])==43: op=1 elif ord(S[i])==45: op=-1 print(Rez) </pre>	<pre> static void Main(string[] args){ string s = Console.ReadLine(); int Rez = (int)s[0] - 48; int op = 1; for (int i=1; i<s.Length; i++) { if (s[i] == '+') op = 1; else if (s[i]=='-') op = -1; else Rez = Rez + op * ((int)s[i] - 48); } Console.WriteLine(Rez); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[]args){ Scanner sc=new Scanner(System.in); String S=sc.nextLine(); int op=1, P; P=(int)S.charAt(0)-48; for (int i=1; i<S.length(); i++){ int m=(int)S.charAt(i)-48; if (m==-5) op=1; else if (m==-3) op=-1; else P=P+m*op; } System.out.println(P); } </pre>	<pre> package main import ("fmt" "unicode") func main() { var S string; var Rez, op int=0,1 fmt.Scan(&S) Rez=int(S[0])-48 for i:= 1; i<len(S); i++){ if unicode.IsDigit(rune(S[i])){ Rez=Rez+op*(int(S[i])-48) } else if (int(S[i])== 43){ op = 1 } else if (int(S[i])==45){ op= -1; } } fmt.Println(Rez) } </pre>

3. Обработка строк с помощью стандартных функций. Поиск и замена

В заданиях, связанных с поиском и заменой подстрок, можно считать, что исходная строка не содержит *перекрывающихся* вхождений требуемых подстрок.

9) Дано целое число $N (> 0)$ и строка S . Преобразовать строку S в строку длины N следующим образом: если длина строки S больше N , то отбросить первые символы, если длина строки S меньше N , то в ее начало добавить символы «.» (точка).

PascalABC.NET		C++	
<pre> Begin var N := ReadInteger; var S := Readstring; if length(S) > N then delete(S,1,length(S)-N) else if length(S) < N then for var i:=length(S) to N-1 do s := '.' + s; writeln(S); end. </pre>		<pre> #include <iostream> #include <string> using namespace std; int main() { string S; getline(cin, S); int n, i, m; cin >> n; m = S.size(); if (m > n) S=S.substr(m - n, m); else S = string(n - m, '*') + S; cout << S<<endl; system("pause"); return 0; } </pre>	
Python		C#	
<pre> n=int(input("n->")) S=input("S->") S=S.strip() #удалить лишние пробелы m=len(S) if m>n: S=S[m-n:] else: for i in range (m,n): S="."+S print(S) </pre>		<pre> static void Main(string[] args) { string S = Console.ReadLine(); int n=Convert.ToInt32(Console.ReadLine()); int m = S.Length; if (m > n) S = S.Substring(m - n, n); else S =new string('*', (n - m)) + S; Console.WriteLine(S); Console.ReadKey(); } </pre>	
Java		Go	
<pre> public static void main(String[]args){ Scanner sc=new Scanner(System.in); String S=sc.nextLine(); int m = S.length(); int n=sc.nextInt(); if (m > n) S=S.substring(m-n, m); else for (int i=0; i<n-m; i++) S="*"+S; System.out.println(S); } </pre>		<pre> package main import "fmt" func main() { var S string; var n,m int fmt.Scan(&n) fmt.Scan(&S) m=len(S) if m>n{ S= S[:m-n] }else{ </pre>	

	<pre> for i:=m; i<n; i++){ S=S+ "." } } fmt.Println(S) } </pre>
--	--

10) Дан символ C и строки S , S_0 . После каждого вхождения символа C в строку S вставить строку S_0 .

PascalABC.NET	C++
<pre> Var S, S0: string; C: Char; i: integer; begin Write('C:'); Readln(C); Write('S:'); Readln(S); Write('S0:'); Readln(S0); i := 1; while(i <= length(S)) do begin if S[i] = C then begin Insert(S0,S,i+1); i := i + Length(S0); end; inc(i); end; writeln(S); end. </pre>	<pre> #include <iostream> #include <string> using namespace std; int main(){ string S, S0; getline(cin, S); getline(cin, S0); unsigned char C; cin >> C; int i = 0; while (i < S.size()) { if (S[i] == C) { S.insert(i+1, S0); i += S0.size(); } else i++; } cout << S << endl; system("pause"); return 0; } </pre>
Python	C#
<pre> c=input("c->") S=input("S->") S0=input("S0->") #удалить лишние пробелы S=S.strip() m=len(S) p=0 S2="" for i in range (0,m): ss="" if (S[i]==c[0]): ss=S[p:i+1]+S0 p=i+1 S2=S2+ss S2=S2+S[p:] print(S2) </pre>	<pre> static void Main(string[] args) { string S = Console.ReadLine(); string S0 = Console.ReadLine(); char c=Convert.ToChar(Console.ReadLine()); int i = 0; while (i < S.Length) { if (S[i] == c) { S=S.Insert(i+1, S0); i += S0.Length+1; } else i++; } Console.WriteLine(S); Console.ReadKey(); } </pre>
Java	Go

<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String S0=sc.nextLine(); char C=sc.nextLine().charAt(0); String S1=""; int n=S.length(); int p=0; for (int i=0; i<n; i++){ if (S.charAt(i)==C){ S1=S1+S.substring(p, i+1)+S0; p=i+1; } } S1=S1+S.substring(p, S.length()); System.out.println(S1); } </pre>	<pre> package main import ("fmt" "os" "bufio") func main() { reader := bufio.NewReader(os.Stdin) var C string fmt.Println("Enter C: ") fmt.Scanf("%s\n",&C) fmt.Println("Enter S: ") S,_:= reader.ReadString('\n') fmt.Println("Enter S0: ") S0,_:=reader.ReadString('\n') S2:="" var i, p int=0,0 for ; i<len(S); i++){ ss:="" if (S[i]==C[0]){ ss=S[p:i+1]+S0 fmt.Println(ss) p=i+1 } S2=S2+ss } S2=S2+S[p:] fmt.Println(S2) } </pre>
--	--

11) Даны строки S и S_0 . Удалить из строки S последнюю подстроку, совпадающую с S_0 . Если совпадающих подстрок нет, то вывести строку S без изменений.

PascalABC.NET	C++
<pre> Var s, s0: string; num, poz, newpoz: integer; begin Write('S: '); Readln(s); Write('S0: '); Readln(s0); poz := 0; repeat newpoz:=pos(s0,copy(s,poz+1,length(s))); if newpoz <> 0 then poz := poz + newpoz; until newpoz = 0; if (poz <> 0) then delete(s, poz, length(s0)); Writeln(s); end. </pre>	<pre> #include <iostream> #include <string> using namespace std; int main(){ string S, S0; getline(cin, S); getline(cin, S0); int pos = S.find_last_of(S0); if (pos+S0.size() < S.size()) S = S.substr(0, pos - 2) + S.substr((pos + S0.size()) - 1, S.size()); else S = S.substr(0, pos - 2); cout << S << endl; system("pause"); return 0; } </pre>

Python	C#
<pre>S=input("S->") S0=input("S0->") S=S.strip() #удалить лишние пробелы m=S.rfind(S0) # посл. подстрока S=S[:m]+S[m+len(S0)+1:] print(S)</pre>	<pre>static void Main(string[] args) { string S = Console.ReadLine(); string S0 = Console.ReadLine(); int pos =S.LastIndexOf(S0); S=S.Remove(pos, S0.Length); Console.WriteLine(S); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String S0=sc.nextLine(); int pos=S.lastIndexOf(S0); String S1=S.substring(0, pos)+ S.substring(pos+S0.length(), S.length()); System.out.println(S1); }</pre>	<pre>package main import ("fmt" "strings" "os" "bufio") func main() { reader := bufio.NewReader(os.Stdin) fmt.Println("Enter S: ") S, _:= reader.ReadString('\n') fmt.Println("Enter S0: ") S1, _ := reader.ReadString('\n') var m,n, k int=0,0,0 n=strings.Index(S,S1) m=n for len(S)>1 && n!=-1{ m=n+ len(S1) S=S[m:] n=strings.Index(S,S1) k++ } if k<=1{ m=m-len(S1) } fmt.Println(m) }</pre>

4. Анализ и преобразование слов в строке

Во всех заданиях данного пункта предполагается, что исходные строки являются непустыми и не содержат начальных и конечных пробелов.

12) Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти количество слов в строке.

PascalABC.NET	C++
<pre>Var S: String; i, Num: integer; begin Write('S: ');</pre>	<pre>#include "stdafx.h" #include <iostream> #include <string> #include <vector> using namespace std;</pre>

<pre> Readln(S); i := 1; Num := 0; while (i <= Length(S)) do begin if (S[i]=' ') and (S[i - 1] <> ' ') then inc(Num); inc(i); end; Writeln(Num + 1); end. Вариант 2. begin WriteLn('Введено слов: ', ReadLnString('Введите текст:').ToWords(' ').Count); end. </pre>	<pre> const vector<string> Split(const string& s, const char& c) { string buff=""; vector<string> v; for (auto n : s) { if (n != c) buff += n; else if (n == c && buff != "") { v.push_back(buff); buff = ""; } } if (buff != "") v.push_back(buff); return v; } int main(){ string S; getline(cin,S); vector<string> v=Split(S, ' '); cout << v.size() << endl; system("pause"); return 0; } </pre>
Python	C#
<pre> S=input("S->") x=S.split(" ") k=0 for i in range(0,len(x)): if x[i]!="": k+=1 print(k) </pre>	<pre> static void Main(string[] args) { string S =Console.ReadLine(); string []x=S.Split(new char[] { ' ' }); Console.WriteLine(x.Count()); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String [] x=S.split(" "); System.out.println(x.length); } </pre>	<pre> package main import ("fmt" "strings" "os" "bufio") func main() { reader := bufio.NewReader(os.Stdin) fmt.Println("Enter S: ") S,_:=reader.ReadString('\n') a := strings.Split(S, " ") fmt.Println(len(a)) } </pre>

13) Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат хотя бы одну букву «А».

PascalABC.NET	C++
<pre> Begin var S:=ReadLnString(S:'); var S1 := S.ToWords(' '); var k := 0; for var i := 0 to S1.Count - 1 do if s1[i].Contains('A') then k += 1; Writeln(k); end. </pre>	<pre> #include <iostream> #include <string> #include <vector> using namespace std; const vector<string> Split(const string& s, const char& c) { string buff=""; vector<string> v; for (auto n : s) { if (n != c) buff += n; else if (n == c && buff != "") { v.push_back(buff); buff = ""; } } if (buff != "") v.push_back(buff); return v; } int main(){ string S; getline(cin,S); int k = 0; vector<string> v=Split(S, ' '); for (auto ss:v) { if ((ss.find("A") >= 0) && (ss.find("A") < ss.size())) k++; } cout << k << endl; system("pause"); return 0; } </pre>
Python	C#
<pre> S=input("S->") x=S.split(" ") k=0 for i in range(0,len(x)): if x[i].find("A")!=-1: k+=1 print(k) </pre>	<pre> static void Main(string[] args) { string S =Console.ReadLine(); string []x=S.Split(new char[] { ' ' }); int k = 0; for (int i=0; i<x.Count(); i++) if (x[i].IndexOf('a') >= 0) k++; Console.WriteLine(k); Console.ReadKey(); } </pre>

Java	Go
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String [] x=S.split(" "); int k=0; for (int i=0; i<x.length; i++) if (x[i].indexOf('A')>=0) k++; System.out.println(k); } </pre>	<pre> package main import ("fmt" "strings" "os" "bufio") func main() { reader := bufio.NewReader(os.Stdin) fmt.Println("Enter S: ") S, _ := reader.ReadString('\n') a := strings.Split(S, " ") var k int=0 for i:=0; i<len(a); i++){ if strings.Contains(a[i], "A"){ k++ } } fmt.Println(k) } </pre>

15) Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все последующие вхождения его первой буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «МИНИ.У.». Количество пробелов между словами не изменять.

PascalABC.NET	C++
<pre> Var S: String; i: integer; C1: Char; begin Write('S: '); Readln(S); C1 := S[1]; i := 2; while (i <= Length(S)) do begin if (S[i - 1] = ' ') and (S[i] <> ' ') then C1 := S[i] else if S[i] = C1 then S[i] := '.'; inc(i); end; Writeln(S); end. </pre>	<pre> #include <iostream> #include <string> #include <vector> using namespace std; const vector<string> Split(const string& s, const char& c){ string buff = ""; vector<string> v; for (auto n : s) { if (n != c) buff += n; else if (n == c && buff != "") { v.push_back(buff); buff = ""; } } if (buff != "") v.push_back(buff); return v; } int main() { </pre>

	<pre> string S; getline(cin, S); int k = 0; vector<string> v = Split(S, ' '); for (string ss : v) { for (int i = 1; i < ss.length(); i++) if (ss[i] == ss[0]) ss.replace(i, 1, "."); cout << ss << endl; } system("pause"); return 0; } </pre>
Python	C#
<pre> S=input("S->") x=S.split(" ") for i in range (0, len(x)): c=x[i][0] ss=x[i][1:]; ss=c+ss.replace(c, ".") print(ss) </pre>	<pre> static void Main(string[] args) { string S =Console.ReadLine(); string []x=S.Split(new char[] { ' ' }); for (int i=0; i<x.Count(); i++){ char c=x[i][0]; string Ss = c.ToString(); for (int j=1; j<x[i].Length; j++) { if (x[i][j]==c) Ss=Ss+ "."; else Ss=Ss+x[i][j]; } x[i]=Ss; } S = ""; for (int j=0; j<x.Count(); j++) { S=S+x[j]+" "; Console.WriteLine(S); Console.ReadKey(); } } </pre>
Java	Go
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String [] x=S.split(" "); int k=0; for (int i=0; i<x.length; i++) { char c=x[i].charAt(0); String ss="" +x[i].charAt(0); System.out.println(c); for (int j=1; j<x[i].length(); j++) if (x[i].charAt(j)==c) ss=ss+ "."; else ss=ss+x[i].charAt(j); x[i]=ss; } } </pre>	<pre> package main import ("fmt" "strings" "os" "bufio") func main() { reader := bufio.NewReader(os.Stdin) fmt.Println("Enter S: ") S, _:=reader.ReadString('\n') a := strings.Split(S, " ") for i:=0; i<len(a); i++){ c:=string(a[i][0]) ss:=a[i][1:]; } } </pre>

<pre> } for (int j=0; j<x.length; j++) System.out.println(x[j]); } </pre>	<pre> ss=c+strings.Replace(ss, c, ".", -1) fmt.Println(ss) } } </pre>
--	---

16) Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке знаков препинания.

PascalABC.NET		C++	
<pre> Begin var S := ReadLnString('Введите текст:'); var k := 0; for var i:=1 to S.Length do if char.IsPunctuation(S[i]) then k += 1; Writeln(k); end. </pre>		<pre> #include <iostream> #include <string> using namespace std; int main() { string S; getline(cin, S); int k = 0; for (auto c: S) { if (ispunct(c)) k++; } cout << k; system("pause"); return 0; } </pre>	
Python		C#	
<pre> import string S=input("S->") k=0 for c in S: if c in string.punctuation: k+=1 print(k) </pre>		<pre> static void Main(string[] args) { string S =Console.ReadLine(); int i, k = 0; for (i=0; i<S.Length; i++) if (Char.IsPunctuation(S[i])) k++; Console.WriteLine(k); Console.ReadKey(); } </pre>	
Java			
<pre> package main import ("fmt" "os" "bufio" "unicode") func main() { reader := bufio.NewReader(os.Stdin) fmt.Println("Enter S: ") S, _ := reader.ReadString('\n') var k int =0 for i:=0; i<len(S); i++){ </pre>			

```

        if unicode.IsPunct(rune(S[i])){
            k++
        }
    }
    fmt.Println(k)
}

```

5. Дополнительные задания на обработку строк

17) Дана строка, содержащая *полное имя файла*, то есть имя диска, список каталогов (путь), собственно имя и расширение. Выделить из этой строки имя файла (без расширения).

PascalABC.NET		C++	
<pre> Begin var S := ReadLnString('Введите текст:'); var S1 := S.ToWords('\ '); S := S1[S1.Count - 1]; var pos := pos('.', S); delete(S,pos,S.Length-pos+1); WriteLn(S); end. </pre>		<pre> #include <iostream> #include <string> using namespace std; int main() { string S; getline(cin, S); int pos = S.find_last_of("/"); S = S.substr(pos + 1, S.size()); pos = S.find("."); S = S.substr(0, pos); cout << S; system("pause"); return 0; } </pre>	
Python	C#		
<pre> import os S=input("S->") x=os.path.basename(S) x=x[:x.find(".")] print(x) </pre>	<pre> static void Main(string[] args) { string S =Console.ReadLine(); int pos=S.LastIndexOf('/'); string Ss=S.Substring(pos+1, S.Length- pos-1); Ss = Ss.Substring(0, Ss.IndexOf(".")); Console.WriteLine(Ss); Console.ReadKey(); } </pre>		
Java		Go	
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); int pos=S.lastIndexOf('/'); int pos1=S.lastIndexOf('.'); String Ss=S.substring(pos+1, pos1); System.out.println(Ss); } </pre>		<pre> package main import ("fmt" "path/filepath" "strings" "bufio" "os") func main() { </pre>	

	<pre> reader := bufio.NewReader(os.Stdin) fmt.Println("Path: ") path, _ := reader.ReadString('\n') file := filepath.Base(path) n:=strings.Index(file, ".") file=file[0:n] fmt.Println(file) } </pre>
--	--

18) Дана строка-предложение на русском языке. Зашифровать ее, выполнив циклическую замену каждой буквы на следующую за ней в алфавите и сохранив при этом регистр букв («А» перейдет в «Б», «а» — в «б», «Б» — в «В», «я» — в «а» и т. д.). Букву «ё» в алфавите не учитывать («е» должна переходить в «ж»). Знаки препинания и пробелы не изменять.

PascalABC.NET	C++
<pre> Begin var S:=ReadLnString('Введите текст:'); for var i := 1 to S.Length do if ord(S[i]) in [1040..1103] then begin var T := ord(S[i]); T := T + 1; if (T=1072) or (T=1104) then T := T - 32; S[i] := chr(T); end; WriteLn(S); end. </pre>	<pre> #include <iostream> #include <string> using namespace std; int main() { string S; getline(cin, S); unsigned char c; for (int i=0; i<S.length(); i++){ c = S[i]; int T = (int)c+1; if (T == 160) T = 128; if (T == 240) T = 160; S[i] = (char)T; } cout << S << endl; system("pause"); return 0; } </pre>
Python	C#
<pre> import os S=input("S->") S1="" for i in range(0, len(S)): t=ord(S[i])+1 if t==1104: t=1072 elif t==1072: t=1040 S1=S1+chr(t) print(S1) </pre>	<pre> static void Main(string[] args) { string S =Console.ReadLine(); string Ss = ""; for (int i=0; i<S.Length; i++){ int T=(int)S[i]; T+=1; if (T == 1104) T = 1072; if (T==1072) T=1040; Ss=Ss+(char)T; } } </pre>

	<pre> Console.WriteLine(Ss); Console.ReadKey(); } </pre>
Java	
<pre> public static void main(String[] args) { Scanner sc=new Scanner(System.in); String S=sc.nextLine(); String Ss=""; for (int i=0; i<S.length(); i++){ int T = (int)S.charAt(i)+1; System.out.println(T); if (T == 1072) T = 1040; if (T == 1104) T = 1072; Ss=Ss+(char)T; } System.out.println(Ss); } </pre>	

- 19) Дана строка, содержащая латинские буквы и скобки трех видов: «()», «[]», «{}». Если скобки расставлены правильно (то есть каждой открывающей соответствует закрывающая скобка того же вида), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная скобка, или, если закрывающих скобок не хватает, число -1.

PascalABC.NET	
<pre> Var i, Num1, Num2, Num3, error: integer; begin var S := ReadLnString('Введите текст:'); i := 1; error := 0; Num1 := 0; Num2 := 0; Num3 := 0; while (i <= Length(S)) do begin if S[i] = '(' then Num1 := Num1 + 1; if S[i] = ')' then Num1 := Num1 - 1; if S[i] = '[' then Num2 := Num2 + 1; if S[i] = ']' then Num2 := Num2 - 1; if S[i] = '{' then Num3 := Num3 + 1; if S[i] = '}' then Num3 := Num3 - 1; if ((Num1<0) or (Num2<0) or (Num3<0)) and (error=0) then error := i; inc(i); end; if error <> 0 then Writeln(error) else if ((Num1 > 0) or (Num2 > 0) or (Num3 > 0)) then Writeln(-1) else Writeln(0); </pre>	

end.

Python

```
import os
S=input("S->")
Num1, Num2, Num3,er,i=0,0,0,0,0
while i<len(S):
    if S[i]=="(":
        Num1+=1
    if S[i]==")":
        Num1-=1
    if S[i]=="[":
        Num2+=1
    if S[i]=="]":
        Num2-=1
    if S[i]=="{":
        Num3+=1
    if S[i]=="}":
        Num3-=1
    if (Num1<0 or Num2<0 or Num3<0) and er==0:
        er=i
    i+=1
if er!=0:
    print(er)
elif (Num1>0) or (Num2>0) or (Num3>0):
    print(-1)
else:
    print(0)
```

Go

```
package main
import (
    "fmt"
    "bufio"
    "os"
    "strings"
)
func main() {
    reader := bufio.NewReader(os.Stdin)
    fmt.Println("Enter S: ")
    S, _ := reader.ReadString('\n')
    var Num1, Num2, Num3,er,i int=0,0,0,0,0
    for i<len(S){
        if strings.Compare(string(S[i]), "(")==0{
            Num1++
        }
        if strings.Compare(string(S[i]), ")")==0{
            Num1--
        }
        if strings.Compare(string(S[i]), "[")==0{
            Num2++
        }
    }
}
```

```

    }
    if strings.Compare(string(S[i]), "]")==0{
        Num2--
    }
    if strings.Compare(string(S[i]), "{")==0{
        Num3++
    }
    if strings.Compare(string(S[i]), "}")==0{
        Num3--
    }
    if (Num1<0 || Num2<0 || Num3<0) && er==0{
        er=i
    }
    i++
}
if er!=0{
    fmt.Println(er)
}else if (Num1>0 || Num2>0 || Num3>0) {
    fmt.Println(-1)
} else{
    fmt.Println(0)
}
}
}

```


Задачи для самостоятельной работы **Символы и их коды. Формирование строк**

- 1) Дан символ C . Вывести два символа, первый из которых предшествует символу C в кодовой таблице, а второй следует за символом C .
- 2) Дано целое число N ($1 \leq N \leq 26$). Вывести N первых *прописных* (то есть заглавных) букв латинского алфавита.
- 3) Дан символ C , изображающий цифру или букву (латинскую или русскую). Если C изображает цифру, то вывести строку «digit», если латинскую букву — вывести строку «lat», если русскую — вывести строку «rus».
- 4) Дана непустая строка. Вывести коды ее первого и последнего символа.
- 5) Дано целое число N (> 0) и символ C . Вывести строку длины N , которая состоит из символов C .
- 6) Дано четное число N (> 0) и символы C_1 и C_2 . Вывести строку длины N , которая состоит из чередующихся символов C_1 и C_2 , начиная с C_1 .
- 7) Дана строка. Вывести строку, содержащую те же символы, но расположенные в обратном порядке.
- 8) Дана непустая строка S и целое число N (> 0). Вывести строку, содержащую символы строки S , между которыми вставлено по N символов «*» (звездочка).

Посимвольный анализ и преобразование строк. **Строки и числа**

- 9) Дана строка. Подсчитать количество содержащихся в ней прописных латинских букв.
- 10) Дана строка. Преобразовать в ней все прописные латинские буквы в строчные.
- 11) Дана строка. Преобразовать в ней все строчные буквы (как латинские, так и русские) в прописные.
- 12) Дана строка. Преобразовать в ней все строчные буквы (как латинские, так и русские) в прописные, а прописные — в строчные.
- 13) Дана строка. Если она представляет собой запись целого числа, то вывести 1, если вещественного (с дробной частью) — вывести 2; если строку нельзя преобразовать в число, то вывести 0. Считать, что дробная часть вещественного числа отделяется от его целой части десятичной *точкой* «.».
- 14) Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке справа налево).
- 15) Дана строка, изображающая целое положительное число. Вывести сумму цифр этого числа.
- 16) Дана строка, изображающая двоичную запись целого положительного числа. Вывести строку, изображающую десятичную запись этого же числа.
- 17) Дана строка, изображающая десятичную запись целого положительного числа. Вывести строку, изображающую двоичную запись этого же числа.

Обработка строк с помощью стандартных функций. Поиск и замена

- 18) Даны целые положительные числа N_1 и N_2 и строки S_1 и S_2 . Получить из этих строк новую строку, содержащую первые N_1 символов строки S_1 и последние N_2 символов строки S_2 (в указанном порядке).

- 19) Дан символ C и строка S . Удвоить каждое вхождение символа C в строку S .
- 20) Дан символ C и строки S, S_0 . Перед каждым вхождением символа C в строку S вставить строку S_0 .
- 21) Даны строки S и S_0 . Проверить, содержится ли строка S_0 в строке S . Если содержится, то вывести True, если не содержится, то вывести False.
- 22) Даны строки S и S_0 . Найти количество вхождений строки S_0 в строку S .
- 23) Даны строки S и S_0 . Удалить из строки S первую подстроку, совпадающую с S_0 . Если совпадающих подстрок нет, то вывести строку S без изменений.
- 24) Даны строки S и S_0 . Удалить из строки S все подстроки, совпадающие с S_0 . Если совпадающих подстрок нет, то вывести строку S без изменений.
- 25) Даны строки S, S_1 и S_2 . Заменить в строке S первое вхождение строки S_1 на строку S_2 .
- 26) Даны строки S, S_1 и S_2 . Заменить в строке S последнее вхождение строки S_1 на строку S_2 .
- 27) Даны строки S, S_1 и S_2 . Заменить в строке S все вхождения строки S_1 на строку S_2 .
- 28) Дана строка, содержащая по крайней мере один символ пробела. Вывести подстроку, расположенную между первым и вторым пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.
- 29) Дана строка, содержащая по крайней мере один символ пробела. Вывести подстроку, расположенную между первым и последним пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.

Анализ и преобразование слов в строке

Во всех заданиях данного пункта предполагается, что исходные строки являются непустыми и не содержат начальных и конечных пробелов.

- 30) Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые начинаются и заканчиваются одной и той же буквой.
- 31) Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат ровно три буквы «А».
- 32) Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого короткого слова.
- 33) Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого длинного слова.
- 34) Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним символом «.» (точка). В конце строки точку не ставить.
- 35) Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все предыдущие вхождения его последней буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «.ИНИ.УМ». Количество пробелов между словами не изменять.

- 36) Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним пробелом и расположенные в обратном порядке.
- 37) Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним пробелом и расположенные в алфавитном порядке.
- 38) Дана строка-предложение на русском языке. Преобразовать строку так, чтобы каждое слово начиналось с заглавной буквы. *Словом* считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки. Слова, не начинающиеся с буквы, не изменять.
- 39) Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке гласных букв.
- 40) Дана строка-предложение на русском языке. Вывести самое длинное слово в предложении. Если таких слов несколько, то вывести первое из них. *Словом* считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки.
- 41) Дана строка-предложение на русском языке. Вывести самое короткое слово в предложении. Если таких слов несколько, то вывести последнее из них. *Словом* считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки.
- 42) Дана строка-предложение с избыточными пробелами между словами. Преобразовать ее так, чтобы между словами был ровно один пробел.

Дополнительные задания на обработку строк

- 43) Дана строка, содержащая *полное имя файла*, то есть имя диска, список каталогов (путь), собственно имя и расширение. Выделить из этой строки расширение файла (без предшествующей точки).
- 44) Дана строка, содержащая полное имя файла. Выделить из этой строки название последнего каталога (без символов «\»). Если файл содержится в корневом каталоге, то вывести символ «\».
- 45) Дана строка-предложение на русском языке и число K ($0 < K < 10$). Зашифровать строку, выполнив циклическую замену каждой буквы на букву того же регистра, расположенную в алфавите на K -й позиции после шифруемой буквы (например, для $K = 2$ «А» перейдет в «В», «а» — в «в», «Б» — в «Г», «я» — в «б» и т. д.). Букву «ё» в алфавите не учитывать, знаки препинания и пробелы не изменять.
- 46) Дано зашифрованное предложение на русском языке (способ шифрования описан в задании String63) и кодовое смещение K ($0 < K < 10$). Расшифровать предложение.

- 47) Дано зашифрованное предложение на русском языке (способ шифрования описан в задании String63) и его расшифрованный первый символ S . Найти кодовое смещение K и расшифровать предложение.
- 48) Дана строка-предложение. Зашифровать ее, поместив вначале все символы, расположенные на четных позициях строки, а затем, в обратном порядке, все символы, расположенные на нечетных позициях (например, строка «Программа» превратится в «ргамамроП»).
- 49) Дано предложение, зашифрованное по правилу. Расшифровать это предложение.
- 50) Дана строка, содержащая цифры и строчные латинские буквы. Если буквы в строке упорядочены по алфавиту, то вывести 0; в противном случае вывести номер первого символа строки, нарушающего алфавитный порядок.
- 51) Дана строка, содержащая латинские буквы и круглые скобки. Если скобки расставлены правильно (то есть каждой открывающей соответствует одна закрывающая), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная закрывающая скобка, или, если закрывающих скобок не хватает, число -1 .

14. Двоичные (типизированные) файлы: группа File

Условие вида «дан файл (целых чисел, вещественных чисел и т. д.)» означает, что в наборе исходных данных указано *имя* данного файла (текстовая строка), а сам исходный файл существует и находится в текущем каталоге. Если существование исходных файлов требуется проверять в ходе выполнения задания, то это особо оговаривается в формулировке задания.

Если в задании требуется создать новый файл, то имя создаваемого файла также входит в набор исходных данных и обычно является *последним* элементом этого набора.

Под размером двоичного типизированного файла всегда подразумевается количество содержащихся в нем *элементов* указанного типа (а не количество байтов, как это принято в операционной системе). В заданиях предполагается, что файловые элементы нумеруются от 1.

Если о минимальном размере исходного файла в задании ничего не сказано, то предполагается, что он равен 2 (то есть файл содержит по крайней мере два элемента). Максимальный размер исходных файлов не устанавливается, поэтому при решении заданий не следует использовать вспомогательные массивы, содержащие все элементы исходных файлов, однако допускается использование *вспомогательных файлов*.

1. Основные операции с двоичными файлами

а) Создание файла, ввод и вывод его элементов

- 1) Дана строка S . Если S является допустимым именем файла, то создать пустой файл с этим именем и вывести True. Если файл с именем S создать нельзя, то вывести False.

PascalABC.NET	C++
<pre> Var S: String; F: file; begin Write('S: '); Readln(S); try Assign(F, S); Rewrite(F); Writeln(TRUE); Close(F); except on e: Exception do writeln(FALSE); end end. </pre>	<pre> #include <iostream> #include <string> #include <fstream> using namespace std; int main(){ fstream file; string S; getline(cin, S); file.open(S, ios_base::out ios_base::in); if (file.is_open()) cout << "Yes"<<endl; else cout << "No"<<endl; system("pause"); } </pre>
Python	C#
<pre> from pathlib import Path my_file = Path(input("FileName:")) print(my_file.is_file()) </pre>	<pre> static void Main(string[] args){ string myFile =Console.ReadLine(); if(File.Exists(@myFile)) Console.WriteLine("Yes"); else Console.WriteLine("No"); Console.ReadKey(); } </pre>

Java	Go
<pre>public static void main(String[] args) { Scanner sc=new Scanner(System.in); String filename=sc.nextLine(); File file=new File(filename); if (file.exists()) System.out.println("Yes"); else System.out.println("No"); }</pre>	<pre>package main import ("fmt" "os") func main() { file, err := os.Create("D://hello.txt") // создаем файл if err != nil{ // если возникла ошибка fmt.Println("Yes:", err) os.Exit(1) // выходим из программы } defer file.Close() // закрываем файл fmt.Println("No") // hello.txt }</pre>
Kotlin	
<pre>import java.util.Scanner import java.io.File fun main(args: Array<String>) { val reader = Scanner(System.`in`) var fileName = reader.next() var fileObject = File(fileName) // create a new file val isNewFileCreated :Boolean = fileObject.createNewFile() if(isNewFileCreated){ print("Yes") } else { print("No") } }</pre>	

2) Дано имя файла и вещественные числа A и D . Создать файл вещественных чисел с данным именем и записать в него 10 первых членов *арифметической прогрессии* с начальным членом A и разностью D :

$$A, A + D, A + 2 \cdot D, A + 3 \cdot D, \dots$$

PascalABC.NET	C++
<pre>Var S: String; F: file of Real; i: integer; A, D, Num: Real; begin Write('S: ');</pre>	<pre>#include <iostream> #include <string> #include <fstream> using namespace std; int main() { string S; float a, d;</pre>

<pre> Readln(S); Write('A,D: '); Readln(A, D); Assign(F, S); ReWrite(F); for i := 0 to 9 do begin Num := A + i * D; Write(F, Num); end; Close(F); end. </pre>	<pre> getline(cin, S); ofstream of(S); if(of.is_open()) { cin >> a >> d; for (int i=0; i<10; i++) of << a+i*d<<" "; of.close(); } else cout << "Unable to open file"; system("pause"); } </pre>
<p style="text-align: center;">Python</p>	<p style="text-align: center;">C#</p>
<pre> import pickle FILENAME = "user.dat" a=float(input("->")) d=float(input("->")) with open(FILENAME, "wb") as file: for i in range(0,10): pickle.dump(a+i*d, file) </pre>	<pre> static void Main(string[] args) { string fileName = @"D:/out.txt"; BinaryWriter writer = new BinaryWriter(File.Open(fileName, FileMode.Create)); double d, a; a=Convert.ToDouble(Console.ReadLine()); d=Convert.ToDouble(Console.ReadLine()); for (int i=0; i<10; i++) writer.Write(a + i * d); writer.Close(); Console.ReadKey(); } </pre>
<p style="text-align: center;">Java</p>	<p style="text-align: center;">Go</p>
<pre> public static void main(String[] args) throws FileNotFoundException, IOException { double a,d; Scanner sc=new Scanner(System.in); a=sc.nextDouble(); d=sc.nextDouble(); String filename = "D:\\input.txt"; RandomAccessFile file=new RandomAccessFile(new File(filename),"rw"); for (int i=0; i<10; i++) file.writeDouble(a+i*d); } </pre>	<pre> package main import ("fmt" "os") func main() { var a,d float64 fmt.Scan(&a,&d) file, err := os.Create("person.dat") if err != nil{ fmt.Println(err) os.Exit(1) } defer file.Close() for i:=0; i<10;i++){ f := float64(i) fmt.Fprintf(file, "%f", a+f*d) } } </pre>

Kotlin
<pre> import java.util.Scanner import java.io.File fun main(args: Array<String>) { val reader = Scanner(System.`in`) var fileName = reader.next() val myfile = File(fileName) var a=reader.nextDouble() var d=reader.nextDouble() myfile.printWriter().use { out -> for (i in (1..10)) out.print((a+i*d).toString()+" ") } println("Writed to file") } </pre>
VB.NET
<pre> Sub Main() Dim a, d As Single Dim writer As BinaryWriter writer = New BinaryWriter(File.Open("D:/output.dat", FileMode.Create)) a = Convert.ToSingle(Console.ReadLine()) d = Convert.ToSingle(Console.ReadLine()) For i As Integer = 0 To 10 writer.Write(a + i * d) Next writer.Close() Console.ReadKey() End Sub </pre>

3) Дано целое число K и файл, содержащий неотрицательные целые числа. Вывести K -й элемент файла (элементы нумеруются от 1). Если такой элемент отсутствует, то вывести -1 .

PascalABC.NET	C++
<pre> Var F: file of Integer; i, Num: integer; begin var S := ReadLnString('Введите имя файла:'); var K := ReadLnInteger(K:'); if fileexists(S) then begin Assign(F, S); Reset(F); Num := 1; while (not eof(F)) and (Num <= K) do begin Read(F, i); inc(Num); </pre>	<pre> #include <iostream> #include <string> #include <fstream> using namespace std; int main() { string S; int k,a; getline(cin, S); ifstream ifs(S); if(ifs.is_open()) { cin >> k; k = k == 2 ? 1 : (k - 2) * sizeof(int); ifs.seekg(k, ios::beg); ifs >> a; cout << a<<" "; } else </pre>

<pre> end; if Num - 1 <> K then Writeln(-1) else writeln(i); Close(F); end; end. </pre>	<pre> cout << "Unable to open file"; system("pause"); } </pre>
<p style="text-align: center;">Python</p>	<p style="text-align: center;">C#</p>
<pre> import pickle FILENAME = "user.dat" k=int(input("k->")) i=0 with open(FILENAME,"rb") as f: while i<k: x=int(pickle.load(f)) i+=1 print(x) </pre>	<pre> static void Main(string[] args){ string fileName = @"D:/out.txt"; BinaryReader reader = new BinaryReader(File.Open(fileName, FileMode.Open)); int k; k=Convert.ToInt32(Console.ReadLine()); reader.BaseStream.Seek(k*sizeof(int), SeekOrigin.Begin); int a = reader.ReadInt32(); Console.WriteLine(a); Console.ReadKey(); } </pre>
<p style="text-align: center;">Java</p>	<p style="text-align: center;">Go</p>
<pre> public static void main(String[] args) throws FileNotFoundException, IOException { int a, k, i=1; Scanner sc=new Scanner(System.in); k=sc.nextInt(); String filename = "D:\\input.txt"; RandomAccessFile file=new RandomAccessFile(new File(filename),"rw"); while (((a=file.readInt())!= 1.0)&& (i<k)) i++; System.out.println(a); } } Варинат 2. public static void main(String[] args) { int a,k; Scanner sc=new Scanner(System.in); k=sc.nextInt(); String filename = "D:\\input.txt"; RandomAccessFile file=new RandomAccessFile(new File(filename),"rw"); file.seek((k-1)*Integer.SIZE/8); a=file.readInt(); System.out.println(a); } } </pre>	<pre> package main import ("fmt" "os") func main() { file, _ := os.Open("input.dat") var x,k int64 fmt.Scan(&k) defer file.Close() file.Seek(k*2,0); fmt.Fscanf(file, "%v", &x) fmt.Printf("%v ",x) } </pre>

Kotlin
<pre> import java.util.Scanner import java.io.File fun main(args: Array<String>) { val reader = Scanner(System.`in`) var fileName = "D:/input.txt"; //reader.next() val myfile = File(fileName) val k = reader.nextInt() var i = 0 var a=0; var readfile = Scanner(myfile); while ((readfile.hasNext()) && (i < k)) { i++; a = readfile.nextInt(); } println(a); } </pre>
VB.NET
<pre> Sub Main() Dim reader As BinaryReader reader = New BinaryReader(File.Open("D:/output.dat", FileMode.Open)) Dim k As Integer = Convert.ToInt32(Console.ReadLine()) reader.BaseStream.Seek((k - 1) * 4, SeekOrigin.Begin) Dim a As Integer = reader.ReadInt32() Console.WriteLine(a) reader.Close() Console.ReadKey() End Sub </pre>

4) Дан файл целых чисел. Создать новый файл, содержащий те же элементы, что и исходный файл, но в обратном порядке.

PascalABC.NET	C++
<pre> Var S: String; F_in, F_out: file of Integer; a: array[1..1000] of real; i,j,R: integer; begin i := 1; Write('File1: '); Readln(S); Assign(F_in, S); Reset(F_in); Write('File2: '); Readln(S); Assign(F_out, S); Rewrite(F_out); while not Eof(F_in) do begin Read(F_in, R); a[i] := R; </pre>	<pre> #include <iostream> #include <string> #include <fstream> #include<vector> using namespace std; int main() { string S; int i,a; getline(cin, S); ifstream ifs(S); ofstream ofs("D:/output.txt"); vector<int>v; if(ifs.is_open()) { while (!ifs.eof()) { ifs >> a; v.push_back(a); } } else </pre>

<pre> i += 1; end; for j:=i-1 downto 1 do Write(F_out, a[j]); Close(F_in); Close(F_out); end. </pre>	<pre> cout << "Unable to open file"; for (i = v.size() - 1; i >= 0; i--) ofs << v[i] << " "; ofs.close(); system("pause"); } </pre>
Python	
<pre> import pickle x = [] with open("D:/input.txt") as f: for line in f: x.append([int(x) for x in line.split()]) with open("D:/output.txt", 'w') as f2: for i in range(len(x[0])-1, 0,-1): f2.write(str(x[0][i])+" ") </pre>	
Go	
<pre> package main import ("fmt" "os" "io") func main() { file, err := os.Open("input.dat") if err != nil{ fmt.Println(err) os.Exit(1) } file1, er1 := os.Create("output.dat") if er1 != nil{ fmt.Println(er1) os.Exit(1) } defer file.Close() defer file1.Close() a:=[100]int{0}; var i int=0 for{ _, err = fmt.Fscanf(file, "%d", &a[i]) if err != nil{ if err == io.EOF{ break } else{ fmt.Println(err) os.Exit(1) } } } i++ } </pre>	

```

    }
    for j:=i-1; j>=0; j--{
        fmt.Fprintf( file,"%d ", a[j])
    }
}

```

5) Дан файл целых чисел. Найти количество содержащихся в нем *серий* (то есть наборов последовательно расположенных одинаковых элементов). Например, для файла с элементами 1, 5, 5, 5, 4, 4, 5 результат равен 4.

PascalABC.NET	C++
<pre> Var S: String; F_in: file of Integer; El, i, temp: integer; begin Assign(F_in, 'D:\INPUT.txt'); Reset(F_in); Read(F_in, temp); El := temp; i := 1; while (not eof(F_in)) do begin Read(F_in, temp); if El <> temp then begin El := temp; inc(i); end; end; Writeln(i); Close(F_in); end. </pre>	<pre> #include <fstream> #include<vector> using namespace std; int main() { string S; int a, b, k=1; getline(cin, S); ifstream ifs(S); ifs >> b; while (!ifs.eof()) { ifs >> a; if (b!= a) { b = a; k++; } } cout << k << " \n "; system("pause"); } </pre>
Python	
<pre> import pickle FILENAME = "input.dat" a=[] with open(FILENAME, "rb") as file: for i in range(0,10): x=pickle.load(file) a.append(x); print(a); a.reverse(); print(a); with open(FILENAME, "wb") as file: for i in range(0,10): pickle.dump(a[i]) </pre>	
Java	Go
<pre> public static void main(String[] args) throws FileNotFoundException, IOException { int a,b, k=0; </pre>	<pre> package main import ("fmt" "os" </pre>

<pre> Scanner sc=new Scanner(System.in); File filename =new File("D:\\input.txt"); Scanner file=new Scanner(filename); b=file.nextInt(); while (file.hasNext()) { a=file.nextInt(); if (b!= a) { b = a; k++; } } System.out.println(k); } </pre>	<pre> "io") func main() { filein, err := os.Open("input.dat") if err != nil{ fmt.Println(err) os.Exit(1) } fileout, er1 := os.Create("output.dat") if er1 != nil{ fmt.Println(er1) os.Exit(1) } defer filein.Close() defer fileout.Close() var a,b int64 var k int=1 fmt.Fscanf(filein, "%d",&a) b=a for{ _, err = fmt.Fscanf(filein, "%d",&a) if err != nil{ if err == io.EOF{ break } else{ fmt.Println(err) os.Exit(1) } } } if b!=a{ b=a k++ } } fmt.Fprintf(fileout,"%d ",k) } </pre>
--	---

б) Дан файл вещественных чисел. Создать файл целых чисел, содержащий длины всех убывающих последовательностей элементов исходного файла. Например, для исходного файла с элементами 1.7, 4.5, 3.4, 2.2, 8.5, 1.2 содержимое результирующего файла должно быть следующим: 3, 2.

PascalABC.NET	C++
<pre> Var F_in: file of Real; F_out: file of Integer; E11, E1: Real; len: integer; </pre>	<pre> #include <iostream> #include <string> #include <fstream> using namespace std; int main(){ </pre>

<pre> begin Assign(F_in, 'D:\input.txt'); Reset(F_in); Assign(F_out, 'D:\output.txt'); ReWrite(F_out); Read(F_in, E1); len := 1; while (not eof(F_in)) do begin E11 := E1; Read(F_in, E1); if E11 > E1 then inc(len) else begin if len > 1 then begin Write(F_out, len); len := 1; end; end; end; Close(F_in); Close(F_out); end. </pre>	<pre> string S, S1; float a, b, k=1; getline(cin, S); ifstream ifs(S); getline(cin, S1); ofstream ofs(S1); ifs >> a; while (!ifs.eof()) { b = a; ifs >> a; if (b > a) { k++; } else if (k > 1) { ofs << k << " "; k = 1; } } system("pause"); } </pre>
Python	C#
<pre> FILENAME = "D:/input.txt" with open(FILENAME) as f: for line in f: x=(float(x) for x in line.split())) m=len(x) len=1 a=x[0] for i in range (1, m- 1): b=a a=x[i] if b>a: len+=1 else: if len>1: with open("D:/output.txt", 'a') as f2: f2.write(str(len)+" ") len=1 </pre>	<pre> static void Main(string[] args) { string fileName = @"D:/out.txt"; string fileName1 = @"D:/out1.txt"; BinaryReader reader = new BinaryReader(File.Open(fileName, FileMode.Open)); BinaryWriter writer = new BinaryWriter(File.Open(fileName1, FileMode.Create)); double a=reader.ReadDouble(); double b; int k=1,pos=sizeof(double); int length = (int)reader.BaseStream.Length; while (pos < length) { b = a; a=reader.ReadDouble(); if (b > a) { k++; } else if (k > 1) { writer.Write(k); k = 1; } } } </pre>

	<pre> pos += sizeof(double); } Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) { double a,b; int k=1; File filename =new File ("D:\\input.txt"); Scanner file=new Scanner(filename); DataOutputStream dos = new DataOutputStream(new FileOutputStream ("D:\\out1.txt")); a=file.nextDouble(); while (file.hasNextDouble()) { b=a; a=file.nextDouble(); if (b > a) { k++; } else if (k > 1) { dos.write(k); k = 1; } } } } </pre>	<pre> package main import ("fmt" "os" "io") func main() { filein, err := os.Open("input.dat") if err != nil{ fmt.Println(err) os.Exit(1) } fileout, er1 := os.Create("output.dat") if er1 != nil{ fmt.Println(er1) os.Exit(1) } defer filein.Close() defer fileout.Close() var a,b float64 var k int=1 fmt.Fscanf(filein, "%f",&a) for{ b=a _, err = fmt.Fscanf(filein, "%f",&a) if err != nil{ if err == io.EOF{ break } else{ fmt.Println(err) os.Exit(1) } } if b>a{ k++ }else{ if k>1{ </pre>

	<pre> fmt.Fprintf(fileout, "%d", k) fmt.Println(k) k=1 } } } } </pre>
--	---

б) Преобразование файла

7) Дан файл вещественных чисел. Заменить в нем все элементы на их квадраты.

PascalABC.NET	C++
<pre> Var F_in, F_temp: file of Real; temp: Real; begin Assign(F_in, 'D:\input.txt'); Reset(F_in); Assign(F_temp, 'D:\input1.txt'); Rewrite(F_temp); while (not eof(F_in)) do begin Read(F_in, temp); temp := sqr(temp); Write(F_temp, temp); end; Close(F_in); Close(F_temp); Erase(F_in); Rename(F_temp, 'D:\input.txt'); end. </pre>	<pre> #include <iostream> #include <string> #include <fstream> #include <vector> using namespace std; int main() { string S; int i, a; getline(cin, S); ifstream ifs(S); vector<int>v; while (!ifs.eof()) { ifs >> a; v.push_back(a); } ifs.close(); ofstream ofs(S); for (i = 0; i<v.size(); i++) ofs << v[i]*v[i]<< " "; ofs.close(); system("pause"); } </pre>
Python	C#
<pre> import pickle FILENAME = "user.dat" f = open(FILENAME, "r+b") s = f.read(8) while s: x = pickle.unpack("d", s) f.seek(-8,1) f.write(pickle.pack("d",x[0]**2)) # Метод flush () очищает внутренний буфер f.flush() s = f.read(8) f.close() </pre>	<pre> static void Main(string[] args) { string fileName = @"D:/out.txt"; string fileName1 = @"D:/out1.txt"; BinaryReader reader = new BinaryReader(File.Open(fileName, FileMode.Open)); BinaryWriter writer = new BinaryWriter(File.Open(fileName1, FileMode.Create)); int pos = 0; double a; </pre>


```

int length =
(int)reader.BaseStream.Length;
while (pos < length){
    a=reader.ReadDouble();
    writer.Write(a*a);
    pos += sizeof(double);
}
reader.Close();
writer.Close();
File.Delete(fileName);
File.Move(fileName1, fileName);
Console.ReadKey();
}

```

Java

```

package main
import (
    "fmt"
    "os"
    "io"
)
func main() {
    filein, err := os.Open("input.dat")
    if err != nil{
        fmt.Println(err)
        os.Exit(1)
    }
    fileout, er1 := os.Create("output.dat")
    if er1 != nil{
        fmt.Println(er1)
        os.Exit(1)
    }
    var a float64
    for{
        _, err = fmt.Fscanf(filein, "%f",&a)
        if err != nil{
            if err == io.EOF{
                break
            } else{
                fmt.Println(err)
                os.Exit(1)
            }
        }
        fmt.Fprintf(fileout, "%f ", a*a)
    }
    filein.Close()
    fileout.Close()
    os.Remove("input.dat");
    os.Rename("output.dat","input.dat")
    os.Remove("output.dat");
}

```

8) Дан файл целых чисел. Удалить из него все элементы с четными номерами.

PascalABC.NET	C++
<pre> Var F_in, F_out: file of Integer; El, i, len: integer; begin Assign(F_in, 'D:\input.txt'); Reset(F_in); Assign(F_out, 'D:\input1.txt'); Rewrite(F_out); i := 1; while not eof(F_in) do begin Read(F_in, El); if odd(i) then Write(F_out, El); inc(i); end; Close(F_in); Close(F_out); Erase(F_in); Rename(F_out, 'D:\input.txt'); end. </pre>	<pre> #include <iostream> #include <string> #include <fstream> #include<vector> using namespace std; int main() { string S; int i, a; getline(cin, S); ifstream ifs(S); vector<int>v; while (!ifs.eof()) { ifs >> a; v.push_back(a); } ifs.close(); ofstream ofs(S); for (i = 0; i < v.size(); i++) if (i % 2 == 0) ofs << v[i] << " "; ofs.close(); system("pause"); } </pre>
Python	
<pre> import pickle FILENAME = "user.dat" f = open(FILENAME, "r+b") s = f.read(8) i=0 while s: x.append(pickle.unpack("d", s)) s = f.read(8) print(x) for j in range(0,len(x)-1): if j%2==1: f.write(pickle.pack("d", x[j][0]**2)) f.flush() f.close() </pre>	
Java	Go
<pre> public static void main(String[] args) throws FileNotFoundException, IOException { File file1 =new File("D:\\input.txt"); File file2 =new File("D:\\output.txt"); FileWriter fw=new FileWriter(file2); </pre>	<pre> package main import ("fmt" "os" "io") func main() { filein, err := os.Open("input.dat") </pre>

<pre> int k=0, a; Scanner fi=new Scanner(file1); BufferedWriter writer = new BufferedWriter(fw); while (fi.hasNextInt()) { k++; a=fi.nextInt(); if (k%2==0) writer.write(String.valueOf(a)+""); } writer.close(); fi.close(); file1.delete(); file2.renameTo(file1); } </pre>	<pre> if err != nil{ fmt.Println(err) os.Exit(1) } var i int=0 fileout, er1 := os.Create("output.dat") if er1 != nil{ fmt.Println(er1) os.Exit(1) } var a int64 for{ _, err=fmt.Fscanf(filein, "%d",&a) if err != nil{ if err == io.EOF{ break } else{ fmt.Println(err) os.Exit(1) } } i++ if i%2==1{ fmt.Fprintf(fileout, "%d ", a) } } filein.Close() fileout.Close() os.Remove("input.dat"); os.Rename("output.dat","input.dat") os.Remove("output.dat"); } </pre>
---	---

с) Работа с нетипизированными двоичными файлами

9) Даны два файла произвольного типа. Поменять местами их содержимое.

PascalABC.NET	C++
<pre> Var Name1, Name2: String; F_1, F_2: file; begin Name1 := 'D:\input.txt'; Name2 := 'D:\input1.txt'; Assign(F_1, Name1); Assign(F_2, Name2); F_1.Rename('D:\temp.txt'); F_2.Rename(Name1); F_1.Rename(Name2); end. </pre>	<pre> #include <iostream> #include <cstdio> using namespace std; int main(){ string S= "D:/input.txt"; string S1 = "D:/input1.txt"; string temp = "D:/temp.txt"; rename(S.c_str(), temp.c_str()); rename(S1.c_str(), S.c_str()); rename(temp.c_str(), S1.c_str()); system("pause"); return 0; } </pre>

Python	C#
<pre>import os FILENAME = "D:/input.txt" FILENAME1 = "D:/input1.txt" os.rename(FILENAME, "D:/temp.txt") os.rename(FILENAME1,FILENAME) os.rename("D:/temp.txt", FILENAME1)</pre>	<pre>static void Main(string[] args) { string fileName = @"D:/out.txt"; string fileName1 = @"D:/out1.txt"; string tmp = @"D:/temp.txt"; File.Move(fileName1, tmp); File.Move(fileName, fileName1); File.Move(tmp,fileName); Console.ReadKey(); }</pre>
Java	Go
<pre>public static void main(String[] args) throws FileNotFoundException, IOException { File file1 =new File("D:\\ input.txt"); File file2 =new File("D:\\ input1.txt"); File temp =new File("D:\\ temp.txt"); file1.renameTo(temp); file2.renameTo(new File("D:\\ input.txt")); temp.renameTo(new File("D:\\ input1.txt")); }</pre>	<pre>package main import "os" func main() { os.Rename("input.dat", "temp.dat") os.Rename("input1.dat", "input.dat") os.Rename("temp.dat", "input1.dat") os.Remove("temp.dat") }</pre>

Замечание. Записать/считать данные в в бинарном файле на C++.

```
1) Записать

#include <iostream>
#include <fstream>
using namespace std;
int main(){
    ofstream ofs;
    int x = 17;
    ofs.open("junk.dat", ios::binary | ios::out);
    for (int i = 2; i <= 20; i += 2) {
        char x = i;
        ofs.write(&x, sizeof(int));
    }
    ofs.close();
    system("pause");
    return 0;
}

2) Считать

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

```

int main()
{
    char x;
    ifstream ifs;
    ifs.open("junk.dat", ios::binary | ios::in);
    while (!ifs.eof()) {
        ifs.read(&x, sizeof(int));
        cout << int(x)<<" ";
    }
    ifs.close();
    system("pause");
    return 0;
}

```

2. Обработка нескольких числовых файлов.

Файлы-архивы

- 15) Даны три файла целых чисел одинакового размера с именами S_A , S_B , S_C и строка S_D . Создать новый файл с именем S_D , в котором чередовались бы элементы исходных файлов с одним и тем же номером:

$A_1, B_1, C_1, A_2, B_2, C_2, \dots$

```

PascalABC.NET

Var
    SA, SB, SC, SD: String;
    FA, FB, FC, FD: file of Integer;
    E1: Integer;

begin
    Write('SA: ');
    Readln(SA);
    Write('SB: ');
    Readln(SB);
    Write('SC: ');
    Readln(SC);
    Write('SD: ');
    Readln(SD);
    Assign(FA, SA);
    Reset(FA);
    Assign(FB, SB);
    Reset(FB);
    Assign(FC, SC);
    Reset(FC);
    Assign(FD, SD);
    Rewrite(FD);
    while not Eof(FA) do
        begin
            Read(FA, E1);
            Write(FD, E1);
            Read(FB, E1);
            Write(FD, E1);

```

```

    Read(FC, E1);
    Write(FD, E1);
end;
Close(FA);
Close(FB);
Close(FC);
Close(FD);
end.

```

Python

```

import pickle
f1 = open("in1.dat", "rb")
f2 = open("in2.dat", "rb")
f3 = open("in3.dat", "rb")
f = open("out.dat", "w+b")
a = f1.read(8)
b= f2.read(8)
c= f3.read(8)
x,y,z=[],[],[]
while a and b and c:
    x.append(pickle.unpack("d", a))
    a = f1.read(8)
    y.append(pickle.unpack("d", b))
    b = f2.read(8)
    z.append(pickle.unpack("d", c))
    c = f3.read(8)
for i in range(0,len(x)-1):
    f.write(pickle.pack('d', x[i][0]))
    f.write(pickle.pack('d', y[i][0]))
    f.write(pickle.pack('d', z[i][0]))

```

Go

```

package main
import (
    "fmt"
    "os"
    "io"
)
func main() {
    filein1, err := os.Open("input.dat")
    filein2, _ := os.Open("input1.dat")
    filein3, _ := os.Open("input2.dat")
    fileout, er1 := os.Create("output.dat")
    if err != nil{
        fmt.Println(err)
        os.Exit(1)
    }
    if er1 != nil{
        fmt.Println(er1)
        os.Exit(1)
    }
}

```

```

}
defer filein1.Close()
defer filein2.Close()
defer filein2.Close()
defer fileout.Close()
var a int64
for{
    _, err = fmt.Fscanf(filein1, "%d",&a)
    if err != nil{
        if err == io.EOF{
            break
        } else{
            fmt.Println(err)
            os.Exit(1)
        }
    }
    }
    fmt.Fprintf(fileout,"%d ",a)
    fmt.Fscanf(filein2, "%d",&a)
    fmt.Fprintf(fileout,"%d ",a)
    fmt.Fscanf(filein3, "%d",&a)
    fmt.Fprintf(fileout,"%d ",a)
}
}

```

3. Символьные и строковые файлы

Строковым файлом называется двоичный типизированный файл, элементами которого являются текстовые строки, хранящиеся в участках памяти *одинакового размера*.

- 16) Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные после первого символа пробела, включая и этот пробел.

PascalABC.NET	C++
<pre> Var S: String; F_in, F_out: file of Char; C: Char; begin Write('File: '); Readln(S); Assign(F_in, S); Reset(F_in); Assign(F_out, 'temp.txt'); Rewrite(F_out); Read(F_in, C); while (not eof(F_in)) and (C <> ' ') do begin Write(F_out, C); Read(F_in, C); end; end; </pre>	<pre> #include <iostream> #include <fstream> #include <string> using namespace std; int main(){ string S = "D:/input.txt"; ifstream file(S); string str; while ((file) && (str.length()>0)) file>>str; file.close(); cout << str << "\n"; ofstream ofile(S,ios::out); ofile << str; ofile.close(); system("pause"); return 0; } </pre>

<pre> Close(F_out); Close(F_in); Erase(F_in); Rename(F_out, S); end. </pre>	
Python	
<pre> f = open("D:/input.txt","r") s = f.readlines() f.close() f = open(f.name,"w") for e in s: f.write(e[:e.find(" ")]+"\n") f.close() </pre>	
Java	
<pre> public static void main(String[] args) throws FileNotFoundException, IOException { File file1 =new File("D:\\input.txt"); File file2 =new File("D:\\output.txt"); FileWriter fo=new FileWriter(file2); Scanner fi=new Scanner(file1); BufferedWriter writer = new BufferedWriter(fo); String Ss=fi.next(); System.out.println(Ss); writer.write(Ss); writer.close(); file1.delete(); file2.renameTo(file1); } </pre>	
Go	
<pre> package main import ("fmt" "os" "io") func main() { filein1, err := os.Open("input.dat") fileout, er1 := os.Create("output.dat") if err != nil{ fmt.Println(err) os.Exit(1) } if er1 != nil{ fmt.Println(er1) os.Exit(1) } defer filein1.Close() defer fileout.Close() var a rune=5 for a!=32{ </pre>	


```

    _, err = fmt.Fscanf(filein1, "%c",&a)
    if err != nil{
        if err == io.EOF{
            break
        } else{
            fmt.Println(err)
            os.Exit(1)
        }
    }
    if a!=32{
        fmt.Fprintf(fileout,"%c",a)
    }
}
os.Rename("output.txt", "input.txt")
os.Remove("output.txt");
}

```

17) Дан строковый файл, содержащий даты в формате «день/месяц/год», причем под день и месяц отводится по две позиции, а под год — четыре (например, «16/04/2001»). Создать два файла целых чисел, первый из которых содержит значения дней, а второй — значения месяцев для дат из исходного строкового файла (в том же порядке).

PascalABC.NET	C++
<pre> Type st = String[100]; var Name, S: St; F_in: file of St; F_out_day, F_out_month: file of integer; temp, error: integer; begin Write('File In: '); Readln(Name); Assign(F_in, Name); Reset(F_in); Write('File Out Day: '); Readln(Name); Assign(F_out_day, Name); Rewrite(F_out_day); Write('File Out Month: '); Readln(Name); Assign(F_out_month, Name); Rewrite(F_out_month); while (not eof(F_in)) do begin Read(F_in, S); Val(S[1] + S[2], temp, error); </pre>	<pre> #include <iostream> #include <fstream> #include <string> using namespace std; int main() { string fileName = "D:/input.txt"; ifstream file(fileName); ofstream ofile("D:/out1.txt", ios::out); ofstream ofile1("D:/out2.txt", ios::out); string str; while (file) { file >> str; ofile<<str.substr(0,2)<<" "; ofile1<<str.substr(3,2)<<" "; } file.close(); ofile.close(); ofile1.close(); system("pause"); return 0; } </pre>

<pre> Write(F_out_day, temp); Val(S[4] + S[5], temp, error); Write(F_out_month, temp); end; Close(F_out_month); Close(F_out_day); Close(F_in); end. </pre>	
Python	
<pre> f = open("D:/input.txt", "r") s = f.readlines() f.close() f1 = open("D:/out1.txt", "w") f2 = open("D:/out2.txt", "w") for e in s: f1.write(e[0:2]+\n") f2.write(e[3:5]+\n") f1.close() f2.close() </pre>	
Go	
<pre> package main import ("fmt" "os" "io" "strings") func main() { filein1, err := os.Open("input1.dat") if err != nil{ fmt.Println(err) os.Exit(1) } fileout, er1 := os.Create("output.dat") fileout1, er2 := os.Create("output1.dat") if er1 != nil{ fmt.Println(er1) os.Exit(1) } if er2 != nil{ fmt.Println(er2) os.Exit(1) } defer filein1.Close() defer fileout.Close() defer fileout1.Close() var s string for { _, err = fmt.Fscanln(filein1, &s) </pre>	

```

        if err != nil{
            if err == io.EOF{
                break
            } else{
                fmt.Println(err)
                os.Exit(1)
            }
        }
        fmt.Fprintf(fileout, "%v ", strings.Split(s, "/")[0])
        fmt.Fprintf(fileout1, "%v ", strings.Split(s, "/")[1])
    }
}

```

4. Использование файлов для работы с матрицами

Матрицей размера $M \times N$ называется прямоугольная таблица чисел, содержащая M строк и N столбцов. Для работы с матрицами обычно используются *двумерные массивы*. Данный пункт посвящен способам обработки матриц, хранящихся в двоичных типизированных файлах на внешних носителях (дисках). Как и в остальных заданиях на обработку файловых данных, при выполнении заданий из данного пункта *не следует* использовать вспомогательные массивы, содержащие все файловые элементы.

В заданиях данного пункта используются дополнительные понятия теории матриц. Приведем определения этих понятий.

Пусть A — матрица размера $M \times N$. Матрица B называется *транспонированной* к матрице A , если она имеет размер $N \neq M$ и ее элементы удовлетворяют следующему соотношению:

$$B_{I,J} = A_{J,I}, \quad I = 1, \dots, N, \quad J = 1, \dots, M.$$

Пусть A — матрица размера $M \times N$, B — матрица размера $N \neq P$. Матрица C называется *произведением* матриц A и B (и обозначается $A \cdot B$), если она имеет размер $M \neq P$ и ее элементы удовлетворяют следующему соотношению:

$$C_{I,J} = A_{I,1} B_{1,J} + A_{I,2} B_{2,J} + \dots + A_{I,N} B_{N,J}, \quad I = 1, \dots, M, \quad J = 1, \dots, P.$$

Квадратная матрица A порядка M называется *верхнетреугольной*, если все ее элементы, лежащие ниже главной диагонали, равны нулю:

$$A_{I,J} = 0, \quad I > J.$$

Квадратная матрица A порядка M называется *нижнетреугольной*, если все ее элементы, лежащие выше главной диагонали, равны нулю:

$$A_{I,J} = 0, \quad I < J.$$

Квадратная матрица A порядка M называется *трехдиагональной*, если равны нулю все ее элементы, не лежащие на главной диагонали и на двух диагоналях, примыкающих к главной:

$$A_{I,J} = 0, \quad |I - J| > 1.$$

- 18) Даны два целых числа I, J и файл вещественных чисел, содержащий элементы квадратной матрицы (по строкам). Вывести элемент матрицы, расположенный в I -й строке и J -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент отсутствует, то вывести 0.

PascalABC.NET

```

Var
  F: file of Real;
  Cell: Real;
  I, J, i2, len: integer;
  Name: String;

begin
  Write('File: ');
  Readln(Name);
  Write('I: ');
  Readln(I);
  Write('J: ');
  Readln(J);
  Assign(F, Name);
  Reset(F);
  len := 0;
  while (not eof(F)) do
  begin
    Read(F, Cell);
    inc(len);
  end;
  len := trunc(sqrt(len));
  Close(F);
  Reset(F);
  for i2 := 1 to len * (i - 1) + j do
    Read(F, Cell);
    Writeln(Cell);
  Close(F);
end.

```

Python	C++
<pre> f = open("D:/input.txt","r") s = f.readlines() f.close() i=int(input("i->")) j=int(input("j->")) n=0 for e in s: n+=1 if n==i: break print(e.split(" ")[j-1]) </pre>	<pre> #include <iostream> #include <vector> #include <string> #include <sstream> #include <fstream> using namespace std; vector<string> split(string str, char d) { vector<string> v; stringstream ss(str); string tok; while (getline(ss, tok, d)) { v.push_back(tok); } return v; } int main() { ifstream ifs("D:/input.txt"); </pre>

	<pre> string S; int i, j, k=0; cin >> i >> j; while (ifs) { getline(ifs, S); if (k == i-1) break; k++; } vector<string> sep=split(S, ' '); cout<<sep[j-1]<<" "<<endl; system("pause"); return 0; } </pre>
Go	
<pre> package main import ("bufio" "fmt" "os" "strings") func main() { f, _ := os.Open("input.txt") defer f.Close() var i, j int var n int=0 var s string fmt.Scan(&i,&j) scanner := bufio.NewScanner(f) for scanner.Scan() { line := scanner.Text() n++ if n>=i{ s=line break } } fmt.Println(strings.Split(s, " ")[j-1]) } </pre>	

Задачи для самостоятельной работы
Основные операции с двоичными файлами
Создание файла, ввод и вывод его элементов

- 1) Дано имя файла и целое число $N (> 1)$. Создать файл целых чисел с данным именем и записать в него N первых положительных четных чисел (2, 4, ...).
- 2) Даны имена четырех файлов. Найти количество файлов с указанными именами, которые имеются в текущем каталоге.
- 3) Дано имя файла целых чисел. Найти количество элементов, содержащихся в данном файле. Если файла с таким именем не существует, то вывести -1 .
- 4) Дан файл целых чисел, содержащий не менее четырех элементов. Вывести первый, второй, предпоследний и последний элементы данного файла.
- 5) Даны имена двух файлов вещественных чисел. Известно, что первый из них существует и является непустым, а второй в текущем каталоге отсутствует. Создать отсутствующий файл и записать в него начальный и конечный элементы существующего файла (в указанном порядке).
- 6) Даны имена двух файлов вещественных чисел. Известно, что один из них (не обязательно первый) существует и является непустым, а другой в текущем каталоге отсутствует. Создать отсутствующий файл и записать в него конечный и начальный элементы существующего файла (в указанном порядке).
- 7) Дан файл вещественных чисел. Создать два новых файла, первый из которых содержит элементы исходного файла с нечетными номерами (1, 3, ...), а второй — с четными (2, 4, ...).
- 8) Дан файл целых чисел. Создать два новых файла, первый из которых содержит четные числа из исходного файла, а второй — нечетные (в том же порядке). Если четные или нечетные числа в исходном файле отсутствуют, то соответствующий результирующий файл оставить пустым.
- 9) Дан файл целых чисел. Создать два новых файла, первый из которых содержит положительные числа из исходного файла (в обратном порядке), а второй — отрицательные (также в обратном порядке). Если положительные или отрицательные числа в исходном файле отсутствуют, то соответствующий результирующий файл оставить пустым.
- 10) Дан файл вещественных чисел. Найти среднее арифметическое его элементов.
- 11) Дан файл вещественных чисел. Найти сумму его элементов с четными номерами.
- 12) Дан файл целых чисел. Создать новый файл целых чисел, содержащий длины всех серий исходного файла (*серией* называется набор последовательно расположенных одинаковых элементов, а *длиной серии* — количество этих элементов). Например, для исходного файла с элементами 1, 5, 5, 5, 4, 4, 5 содержимое результирующего файла должно быть следующим: 1, 3, 2, 1.
- 13) Дан файл вещественных чисел. Найти его первый локальный минимум (*локальным минимумом* называется элемент, который меньше своих соседей).
- 14) Дан файл вещественных чисел. Найти его последний локальный максимум (*локальным максимумом* называется элемент, который больше своих соседей).
- 15) Дан файл вещественных чисел. Найти общее количество его *локальных экстремумов*, то есть локальных минимумов и локальных максимумов.
- 16) Дан файл вещественных чисел. Создать файл целых чисел, содержащий номера всех *локальных максимумов* исходного файла в порядке возрастания.

- 17) Дан файл вещественных чисел. Создать файл целых чисел, содержащий номера всех *локальных экстремумов* исходного файла в порядке убывания.
- 18) Дан файл вещественных чисел. Создать файл целых чисел, содержащий длины всех монотонных последовательностей элементов исходного файла. Например, для исходного файла с элементами 1.7, 4.5, 3.4, 2.2, 8.5, 1.2 содержимое результирующего файла должно быть следующим: 2, 3, 2, 2.

Преобразование файла

- 19) Дан файл вещественных чисел. Поменять в нем местами минимальный и максимальный элементы.
- 20) Дан файл целых чисел с элементами A_1, A_2, \dots, A_N (N — количество элементов в файле). Заменить исходное расположение его элементов на следующее:
 $A_1, A_N, A_2, A_{N-1}, A_3, \dots$
- 21) Дан файл вещественных чисел. Заменить в файле каждый элемент, кроме начального и конечного, на его среднее арифметическое с предыдущим и последующим элементом.
- 22) Дан файл целых чисел, содержащий более 50 элементов. Уменьшить его размер до 50 элементов, удалив из файла необходимое количество конечных элементов.
- 23) Дан файл целых чисел, содержащий четное количество элементов. Удалить из данного файла вторую половину элементов.
- 24) Дан файл целых чисел, содержащий более 50 элементов. Уменьшить его размер до 50 элементов, удалив из файла необходимое количество начальных элементов.
- 25) Дан файл целых чисел, содержащий четное количество элементов. Удалить из данного файла первую половину элементов.
- 26) Дан файл целых чисел. Удалить из него все отрицательные числа.
- 27) Дан файл целых чисел, содержащий менее 50 элементов. Увеличить его размер до 50 элементов, записав в начало файла необходимое количество нулей.
- 28) Дан файл целых чисел. Продублировать в нем все элементы с нечетными номерами.
- 29) Дан файл целых чисел. Удвоить его размер, записав в конец файла все его исходные элементы (в обратном порядке).
- 30) Дан файл целых чисел. Продублировать в нем все числа, принадлежащие диапазону 5–10.
- 31) Дан файл целых чисел. Заменить в нем каждый элемент с четным номером на два нуля.
- 32) Дан файл целых чисел. Заменить в нем каждое положительное число на три нуля.

Работа с нетипизированными двоичными файлами

- 33) Дан файл произвольного типа. Создать его копию с новым именем.
- 34) Даны три файла одного и того же типа, но разного размера. Заменить содержимое самого длинного файла на содержимое самого короткого.
- 35) Даны три файла одного и того же типа, но разного размера. Заменить содержимое самого короткого файла на содержимое самого длинного.
- 36) Дана строка S_0 , целое число $N (\leq 4)$ и N файлов одного и того же типа с именами S_1, \dots, S_N . Объединить содержимое этих файлов (в указанном порядке) в новом файле с именем S_0 .

37) Даны два файла одного и того же типа. Добавить к первому файлу содержимое второго файла, а ко второму файлу — содержимое первого.

Обработка нескольких числовых файлов.

Файлы-архивы

- 38) Даны четыре файла целых чисел разного размера с именами S_A , S_B , S_C , S_D и строка S_E . Создать новый файл с именем S_E , в котором чередовались бы элементы исходных файлов с одним и тем же номером «Лишние» элементы более длинных файлов в результирующий файл не записывать.
- 39) Даны два файла вещественных чисел с именами S_1 и S_2 , элементы которых упорядочены по возрастанию. Объединить эти файлы в новый файл с именем S_3 так, чтобы его элементы также оказались упорядоченными по возрастанию.
- 40) Даны три файла вещественных чисел с именами S_1 , S_2 и S_3 , элементы которых упорядочены по убыванию. Объединить эти файлы в новый файл с именем S_4 так, чтобы его элементы также оказались упорядоченными по убыванию.
- 41) Дана строка S_0 , целое число $N (\leq 4)$ и N файлов целых чисел с именами S_1, \dots, S_N . Объединить их содержимое в новом файле-*архиве* с именем S_0 , используя следующий формат: в первом элементе файла-архива хранится число N , в следующих N элементах хранится размер (число элементов) каждого из исходных файлов, а затем последовательно размещаются данные из каждого исходного файла.
- 42) Дана строка S , целое число $N (> 0)$ и файл-*архив* целых чисел, содержащий данные из нескольких файлов. Восстановить из файла-архива файл с номером N и сохранить его под именем S . Если файл-архив содержит данные из менее чем N файлов, то оставить результирующий файл пустым.
- 43) Дана строка S и файл-*архив* целых чисел, содержащий данные из нескольких (не более шести) файлов. Для каждого из файлов, содержащихся в архиве, найти среднее арифметическое всех его элементов (вещественное число) и записать найденные числа (в том же порядке) в файл вещественных чисел с именем S .
- 44) Дана строка S_0 , целое число $N (\leq 4)$ и N файлов целых чисел с именами S_1, \dots, S_N . Объединить их содержимое в новом файле-*архиве* с именем S_0 , последовательно записывая в него следующие данные: размер (число элементов) первого исходного файла и все элементы этого файла, размер второго исходного файла и все его элементы, ..., размер N -го исходного файла и все его элементы.
- 45) Дана строка S , целое число $N (> 0)$ и файл-*архив* целых чисел, содержащий данные из нескольких файлов. Восстановить из файла-архива файл с номером N и сохранить его под именем S . Если файл-архив содержит данные из менее чем N файлов, то оставить результирующий файл пустым.
- 46) Даны строки S_1 , S_2 и файл-*архив* целых чисел, содержащий данные из нескольких файлов. Создать новые файлы целых чисел с именами S_1 и S_2 и записать в первый из них начальные элементы всех файлов, содержащихся в архиве, а во второй — конечные элементы этих файлов (в том же порядке).

Символьные и строковые файлы

Строковым файлом называется двоичный типизированный файл, элементами которого являются текстовые строки, хранящиеся в участках памяти *одинакового размера*.

- 47) Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные после последнего символа пробела, включая и этот пробел.
- 48) Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные перед первым символом пробела, включая и этот пробел.
- 49) Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные перед последним символом пробела, включая и этот пробел.
- 50) Дан символьный файл. Упорядочить его элементы по возрастанию их кодов.
- 51) Дано целое число $K (> 0)$ и строковый файл. Создать два новых файла: строковый, содержащий первые K символов каждой строки исходного файла, и символьный, содержащий K -й символ каждой строки (если длина строки меньше K , то в строковый файл записывается вся строка, а в символьный файл записывается пробел).
- 52) Дан строковый файл. Создать новый строковый файл, содержащий все строки исходного файла наименьшей длины (в том же порядке).
- 53) Дан строковый файл. Создать новый строковый файл, содержащий все строки исходного файла наибольшей длины (в обратном порядке).
- 54) Дан строковый файл. Создать новый строковый файл, в котором строки из исходного файла располагались бы в *лексикографическом* порядке, то есть по возрастанию кодов их символов, начиная с первого символа.
- 55) Дан строковый файл, содержащий даты в формате, описанном в задании File67. Создать два файла целых чисел, первый из которых содержит значения месяцев, а второй — значения лет для дат из исходного строкового файла (в обратном порядке).
- 56) Дан строковый файл, содержащий даты в формате, описанном в задании File67. Создать новый строковый файл, содержащий все зимние даты из исходного файла (в обратном порядке). Если даты с требуемым временем года в файле отсутствуют, то оставить результирующий файл пустым.
- 57) Дан строковый файл, содержащий даты в формате, описанном в задании File67. Найти строку, содержащую самую позднюю осеннюю дату. Если даты с требуемым временем года в файле отсутствуют, то вывести пустую строку.
- 58) Дан строковый файл, содержащий даты в формате, описанном в задании File67. Создать новый строковый файл, в котором даты из исходного файла располагались бы в порядке убывания.

Использование файлов для работы с матрицами

В заданиях данного пункта используются дополнительные понятия теории матриц. Приведем определения этих понятий.

Пусть A — матрица размера $M \times N$. Матрица B называется *транспонированной* к матрице A , если она имеет размер $N \neq M$ и ее элементы удовлетворяют следующему соотношению:

$$B_{I,J} = A_{J,I}, \quad I = 1, \dots, N, \quad J = 1, \dots, M.$$

Пусть A — матрица размера $M \times N$, B — матрица размера $N \neq P$. Матрица C называется *произведением* матриц A и B (и обозначается $A \cdot B$), если она имеет размер $M \neq P$ и ее элементы удовлетворяют следующему соотношению:

$$C_{I,J} = A_{I,1} B_{1,J} + A_{I,2} B_{2,J} + \dots + A_{I,N} B_{N,J}, \quad I = 1, \dots, M, \quad J = 1, \dots, P.$$

Квадратная матрица A порядка M называется *верхнетреугольной*, если все ее элементы, лежащие ниже главной диагонали, равны нулю:

$$A_{I,J} = 0, \quad I > J.$$

Квадратная матрица A порядка M называется *нижнетреугольной*, если все ее элементы, лежащие выше главной диагонали, равны нулю:

$$A_{I,J} = 0, \quad I < J.$$

Квадратная матрица A порядка M называется *трехдиагональной*, если равны нулю все ее элементы, не лежащие на главной диагонали и на двух диагоналях, примыкающих к главной:

$$A_{I,J} = 0, \quad |I - J| > 1.$$

- 59) Дан файл вещественных чисел, содержащий элементы квадратной матрицы (по строкам). Создать новый файл, содержащий элементы матрицы, транспонированной к исходной.
- 60) Даны два файла вещественных чисел с именами S_A и S_B , содержащие элементы квадратных матриц A и B (по строкам). Создать новый файл с именем S_C , содержащий элементы произведения $A \cdot B$. Если матрицы A и B нельзя перемножать, то оставить файл S_C пустым.
- 61) Даны два целых числа I, J и файл вещественных чисел, содержащий элементы прямоугольной матрицы (по строкам), причем первый элемент файла содержит количество столбцов матрицы. Вывести элемент матрицы, расположенный в I -й строке и J -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент отсутствует, то вывести 0.
- 62) Дан файл вещественных чисел, содержащий элементы прямоугольной матрицы (по строкам), причем первый элемент файла содержит количество столбцов матрицы. Создать новый файл той же структуры, содержащий матрицу, транспонированную к исходной.
- 63) Даны два файла вещественных чисел с именами S_A и S_B , содержащие элементы прямоугольных матриц A и B (по строкам), причем первый элемент каждого файла содержит количество столбцов соответствующей матрицы. Создать файл той же структуры с именем S_C , содержащий элементы произведения $A \cdot B$. Если матрицы A и B нельзя перемножать, то оставить файл S_C пустым.
- 64) Дан файл вещественных чисел, содержащий элементы верхнетреугольной матрицы (по строкам). Создать новый файл, содержащий элементы ненулевой части данной матрицы (по строкам).
- 65) Дан файл вещественных чисел, содержащий элементы нижнетреугольной матрицы (по строкам). Создать новый файл, содержащий элементы ненулевой части данной матрицы (по строкам).
- 66) Дан файл вещественных чисел, содержащий элементы трехдиагональной матрицы (по строкам). Создать новый файл, содержащий элементы ненулевой части данной матрицы (по строкам).
- 67) Даны два целых числа I, J и файл вещественных чисел, содержащий ненулевую часть верхнетреугольной матрицы (по строкам). Вывести порядок матрицы и ее элемент, расположенный в I -й строке и J -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент находится в нулевой части матрицы, то вывести 0; если элемент отсутствует, то вывести -1 .

- 68) Даны два целых числа I, J и файл вещественных чисел, содержащий ненулевую часть нижнетреугольной матрицы (по строкам). Вывести порядок матрицы и ее элемент, расположенный в I -й строке и J -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент находится в нулевой части матрицы, то вывести 0; если элемент отсутствует, то вывести -1 .
- 69) Даны два целых числа I, J и файл вещественных чисел, содержащий ненулевую часть трехдиагональной матрицы (по строкам). Вывести порядок матрицы и ее элемент, расположенный в I -й строке и J -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент находится в нулевой части матрицы, то вывести 0; если элемент отсутствует, то вывести -1 .
- 70) Дан файл вещественных чисел, содержащий ненулевую часть верхнетреугольной матрицы (по строкам). Создать новый файл, содержащий все элементы данной матрицы (по строкам).
- 71) Дан файл вещественных чисел, содержащий ненулевую часть нижнетреугольной матрицы (по строкам). Создать новый файл, содержащий все элементы данной матрицы (по строкам).
- 72) Дан файл вещественных чисел, содержащий ненулевую часть трехдиагональной матрицы (по строкам). Создать новый файл, содержащий все элементы данной матрицы (по строкам).
- 73) Даны два файла вещественных чисел с именами S_A и S_B , содержащие ненулевые части верхнетреугольных матриц A и B (по строкам). Создать новый файл с именем S_C , содержащий ненулевую часть произведения $A \cdot B$ (по строкам). Если матрицы A и B нельзя перемножить, то оставить файл S_C пустым.
- 74) Даны два файла вещественных чисел с именами S_A и S_B , содержащие ненулевые части нижнетреугольных матриц A и B (по строкам). Создать новый файл с именем S_C , содержащий ненулевую часть произведения $A \cdot B$ (по строкам). Если матрицы A и B нельзя перемножить, то оставить файл S_C пустым.

15. Текстовые файлы: группа Text

Условие «дан текстовый файл» означает, что в наборе исходных данных указано *имя* данного файла (текстовая строка). Все исходные файлы в заданиях данной группы считаются существующими.

Если в задании требуется создать новый файл, то имя создаваемого файла также входит в набор исходных данных (и, как правило, является последним элементом этого набора).

Максимальный размер исходных файлов не устанавливается, поэтому при решении заданий не следует использовать вспомогательные массивы, содержащие все элементы исходных файлов, однако допускается использование *вспомогательных файлов*.

Используемые в заданиях двоичные файлы удовлетворяют условиям, которые перечислены в начале раздела «Двоичные (типизированные) файлы».

1. Основные операции с текстовыми файлами

- 1) Дано имя файла и целые положительные числа N и K . Создать текстовый файл с указанным именем и записать в него N строк, каждая из которых состоит из K символов «*» (звездочка).

PascalABC.NET	C++
<pre> Var F: Text; i, j, N, K: integer; Name: String; begin Write('File name: '); Readln(Name); Write('N,K: '); Readln(N, K); Assign(F, Name); Rewrite(F); for i := 1 to N do begin for j := 1 to K do F.Write('*'); if i <> N then F.Writeln; end; F.Close; end. </pre>	<pre> #include <iostream> #include <string> #include <fstream> using namespace std; int main() { ofstream ofs ("D:/output.txt"); string S; int n, k, i; cin >> n >> k; S = S.append(k, '*'); for (i = 0; i < n; i++) { ofs << S << "\n"; } system("pause"); return 0; } </pre>
Python	C#
<pre> f =open("D:/out.txt","w") n=int(input("n->")) k=int(input("k->")) for i in range(0,n): S="" for j in range(0, k): S=S+ "*" S=S+"\n" f.write(S) f.close() </pre>	<pre> static void Main(string[] args) { string fileName = @"D:/out.txt"; StreamWriter sw = new StreamWriter(fileName); int n, k, i; n=Convert.ToInt32(Console.ReadLine()); k = Convert.ToInt32(Console.ReadLine()); string S=new string('*',k); for (i=0; i<n; i++) sw.WriteLine(S); sw.Close(); Console.ReadKey(); } </pre>

Java	Go
<pre> public static void main(String[] args) throws IOException{ FileWriter writer = new FileWriter("D:\\output.txt"); Scanner sc=new Scanner(System.in); int n, k, i; n=sc.nextInt(); k=sc.nextInt(); String S=""; for (i=0; i<k; i++) S=S+"*"; for (i=0; i<n; i++) writer.write(S+"\n"); writer.close(); } </pre>	<pre> package main import ("bufio" "os" "fmt") func main() { var k, n int fmt.Scan(&k) fmt.Scan(&n) var s string="" for i:=1; i<=k; i++){ s=s+"* " } s=s+string("\n") f, _ := os.Create("file.txt") w := bufio.NewWriter(f) for i:=1; i<=n; i++){ w.WriteString(s) } w.Flush() } </pre>
Kotlin	
<pre> import java.util.Scanner import java.io.File fun main(args: Array<String>) { val reader = Scanner(System.`in`) var fileName = reader.next() val myfile = File(fileName) var n = reader.nextInt() var k = reader.nextInt() var S = "" for (i in (1..k)) S = S + "*"; myfile.printWriter().use { out -> for (i in (1..n)) out.println(S) } } </pre>	

2) Дана строка S и текстовый файл. Добавить строку S в конец файла.

PascalABC.NET	C++
<pre> Var F: Text; Name, S: String; begin Write('File name: '); ReadLn(Name); Write('S: '); </pre>	<pre> #include <iostream> #include <fstream> #include <string> using namespace std; int main() { string S; getline(cin, S); </pre>

<pre> Readln(S); Assign(F, Name); Append(F); F.WriteLine(S); F.Close; end. </pre>	<pre> ofstream ofs("D:\input.txt", ios::app); ofs << S; return 0; } </pre>
Python	C#
<pre> S=input() with open("D:\hello.txt", "a") as file: file.write(S) </pre>	<pre> static void Main(string[] args) { string fileName = @"D:/out.txt"; StreamWriter sw = new StreamWriter (fileName,true); String S=Console.ReadLine(); sw.WriteLine(S); sw.Close(); Console.ReadKey(); } </pre>
Java	Go
<pre> public static void main(String[] args) throws IOException{ FileWriter writer = new FileWriter("D:\\output.txt", true); Scanner sc=new Scanner(System.in); String S=sc.nextLine(); writer.write(S); writer.close(); } </pre>	<pre> package main import ("fmt" "os") func main() { f, err := os.OpenFile("file.txt", os.O_APPEND os.O_WRONLY, os.ModeAppend) if err != nil { fmt.Println(err) } defer f.Close() var s string fmt.Scan(&s) _, err = f.WriteString(s) if err != nil { fmt.Println(err) } } </pre>
Kotlin	
<pre> import java.util.Scanner import java.io.File import java.io.FileWriter fun main(args: Array<String>) { val reader = Scanner(System.`in`) var fileName = reader.nextLine() val myfile = File(fileName) var S=reader.nextLine() val fw = FileWriter(myfile, true) fw.write(S) fw.close() } </pre>	

Считать файл по строкам

```

1)
import java.util.Scanner
import java.io.File
import java.io.FileWriter
fun main(args: Array<String>) {
    val reader = Scanner(System.`in`)
    var fileName = reader.nextLine()
    File(fileName).readLines().forEach {
        println(it)
    }
}

2)
import java.util.Scanner
import java.io.File
import java.io.FileWriter
fun main(args: Array<String>) {
    val reader = Scanner(System.`in`)
    var fileName = reader.nextLine()
    val file = File(fileName)
    val bufferedReader = file.bufferedReader()
    val text: List<String> = bufferedReader.readLines()
    for(line in text){
        println(line)
    }
}

3)
import java.util.Scanner
import java.io.File
import java.io.FileWriter
fun main(args: Array<String>) {
    val reader = Scanner(System.`in`)
    var fileName = reader.nextLine()
    val file = File(fileName)
    var content: String = file.readText()
    println(content)
}

```

3) Дан текстовый файл. Продублировать в нем все пустые строки.

PascalABC.NET	C++
<pre> Var F_in, F_out: Text; Name, S: String; begin Write('File name: '); Readln(Name); Assign(F_in, Name); Assign(F_out, 'D:\input1.txt'); Reset(F_in); Rewrite(F_out); while not F_in.Eof do begin </pre>	<pre> #include <iostream> #include <fstream> #include <string> using namespace std; int main() { string file1 = "D:\input.txt"; string file2 = "D:\input1.txt"; string S; ifstream ifs(file1); ofstream ofs(file2); </pre>

<pre> S := F_in.ReadString; if S = '' then F_out.WriteLine; F_out.WriteLine(S); end; F_in.Close; F_out.Close; Erase(F_in); F_out.Rename(Name); end. </pre>	<pre> while (!ifs.eof()) { getline(ifs, S); if (S.length()== 0) ofs << S << endl; ofs << S << endl; } ifs.close(); ofs.close(); remove(file1.c_str()); rename(file2.c_str(), file1.c_str()); system("pause"); return 0; } </pre>
Python	C#
<pre> f=open("D:/input.txt","r") s=f.readlines() f.close() f=open("D:/input.txt","w") for e in s: if e==" ": f.write(e+"\n") f.write(e+"\n") f.close() </pre>	<pre> static void Main(string[] args) { string fileName = @"D:/out.txt"; StreamReader sr = new StreamReader (fileName); string line; List<string> list = new List <string>(); while((line=sr.ReadLine()) != null) { if (line.Length==0) list.Add("\n\n"); else list.Add(line); } sr.Close(); StreamWriter sw = new StreamWriter (fileName,false); foreach (string x in list){ sw.WriteLine(x); Console.WriteLine(x); } sw.Close(); Console.ReadKey(); } </pre>
Java	
<pre> public static void main(String[] args) throws IOException{ StringBuilder contentBuilder = new StringBuilder(); BufferedReader br = new BufferedReader(new FileReader ("D:\\output.txt")); String sCurrentLine; while ((sCurrentLine = br.readLine()) != null){ if (sCurrentLine.isEmpty()) contentBuilder.append("\n\n"); </pre>	


```

        else
            contentBuilder.append(sCurrentLine).append("\n");
    }
    br.close();
    FileWriter writer = new FileWriter("D:\\output.txt", false);
    writer.write(contentBuilder.toString());
    writer.close();
}

```

Go

```

package main
import (
    "bufio"
    "fmt"
    "os"
)
func main() {
    file, err := os.Create("file.txt")
    if err != nil{
        fmt.Println(err)
        os.Exit(1)
    }
    defer file.Close()
    f, _ := os.Open("input.txt")
    defer f.Close()
    scanner := bufio.NewScanner(f)
    for scanner.Scan() {
        line := scanner.Text()
        fmt.Println(line)
        if len(line)==0{
            fmt.Fprintf( file,"%s \n",line)
        }
        fmt.Fprintf( file,"%s \n",line)
    }
    os.Rename("file.txt","input.dat")
    os.Remove("file.txt")
}

```

- 4) Даны два текстовых файла. Добавить в конец каждой строки первого файла соответствующую строку второго файла. Если второй файл короче первого, то оставшиеся строки первого файла не изменять.

PascalABC.NET

```

Var
    F1_in, F2_in, F_out: Text;
    Name1, Name2, line: string;

begin
    Write('File1 name: ');
    Readln(Name1);
    Write('File2 name: ');

```

```

Readln(Name2);
Assign(F1_in, Name1);
Assign(F2_in, Name2);
Assign(F_out, 'D:\temp.txt');
Reset(F1_in);
Reset(F2_in);
Rewrite(F_out);
while not F1_in.eof do
begin
    line := F1_in.ReadString;
    F_out.Write(line);
    if not F2_in.eof then
        line := F2_in.ReadString
    else
        line := '';
    F_out.Writeln(line);
end;
F1_in.Close;
F2_in.Close;
F_out.Close;
F1_in.Erase;
F_out.Rename(Name1);
end.

```

Python

```

f1 = open("D:/input.txt","r")
s1 = f1.readlines()
f2 = open("D:/input1.txt","r")
s2 = f2.readlines()
f1.close()
f2.close()
f = open("D:/input2.txt","w")
for i in range(0, len(s1)):
    if i<len(s2):
        f.write(s1[i]+s2[i])
    else:
        f.write(s1[i])
f.close()

```

Go

```

package main
import (
    "bufio"
    "fmt"
    "os"
)
func main() {
    f1, _ := os.Open("input.txt")
    f2, _ := os.Open("input1.txt")
    f, _ := os.Create("file.txt")

```

```

defer f1.Close()
defer f2.Close()
var s,s1,s2 string
scanner1 := bufio.NewScanner(f1)
scanner2 := bufio.NewScanner(f2)
for scanner1.Scan() {
    s1 = scanner1.Text()
    if scanner2.Scan() {
        s2 = scanner2.Text()
    }else{
        s2=""
    }
    s=s1+" "+s2
    fmt.Fprintf( f,"%s \n",s)
}
os.Remove("input.txt")
os.Rename("file.txt","input.txt")
os.Remove("file.txt")
}

```

2. Анализ и форматирование текста

5) Дан текстовый файл. Найти количество абзацев в тексте, если абзацы отделяются друг от друга одной или несколькими пустыми строками.

PascalABC.NET	C++
<pre> var F: Text; S: string; num: integer; begin Write('File name: '); Readln(S); Assign(F, S); Reset(F); num := 0; S := ''; while (not F.eof) and (S='') do S := F.ReadlnString; while not F.eof do begin S := F.ReadlnString; if S = '' then begin inc(num); end; while (not F.eof) and (S='') do S := F.ReadlnString; end; end; if S <> '' then inc(num); F.Close; Writeln(num); end. </pre>	<pre> #include <iostream> #include <fstream> #include <string> using namespace std; int main() { string file1 = "D:\input.txt"; string S; ifstream ifs(file1); int k = 0; while (!ifs.eof()) { getline(ifs, S); if (S.length() != 0) k++; } cout << k << endl; system("pause"); return 0; } </pre>

Python
<pre>f = open("D:/input.txt","r") s = f.readlines() f.close() k=0 for e in s: if e!="\n": k+=1 print(k)</pre>
Go
<pre>package main import ("bufio" "fmt" "os") func main() { f, _ := os.Open("input.txt") defer f.Close() var num int=0 scanner := bufio.NewScanner(f) for scanner.Scan() { s:= scanner.Text() if (s!=""){ num++ } } fmt.Println(num) }</pre>

б) Дан текстовый файл. Найти количество абзацев в тексте, если первая строка каждого абзаца начинается с 5 пробелов («красная строка»). Пустые строки между абзацами не учитывать.

PascalABC.NET	C++
<pre>Var F: Text; Name, S: string; num, K: integer; begin Write('File name: '); Readln(Name); Assign(F, Name); Reset(F); num := 0; while not F.eof do begin S := F.ReadlnString; if pos(' ', S) = 1 then inc(num);</pre>	<pre>#include <iostream> #include <fstream> #include <string> using namespace std; int main() { string file1 = "D:\input.txt"; string S; ifstream ifs(file1); int k = 0; while (!ifs.eof()) { getline(ifs, S); if ((S.length() != 0) && (S.find(" ") == 0)) k++; }</pre>

<pre> end; F.Close; Writeln(num); end.</pre>	<pre> cout << k << endl; system("pause"); return 0; }</pre>
Python	
<pre> f = open("D:/input.txt","r") s = f.readlines() f.close() k=0 for e in s: if e.find(" ")==0: k+=1 print(k)</pre>	
Go	
<pre> package main import ("bufio" "fmt" "os" "strings") func main() { f, _ := os.Open("input.txt") defer f.Close() var num int=0 scanner := bufio.NewScanner(f) for scanner.Scan() { s:= scanner.Text() if strings.Index(s, " ")==0{ num++ } } fmt.Println(num) }</pre>	

- 7) Дан текстовый файл. Вывести первое слово текста наибольшей длины. *Словом* считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки.

PascalABC.NET
<pre> Var F: Text; Name, S, Word: string; begin Write('File name: '); Readln(Name); Assign(F, Name); Reset(F); Word := ''; while not F.eof do</pre>

```

begin
  S := F.ReadLnString;
  while (pos(' ', S) <> 0) do
  begin
    if (pos(' ', S) - 1) > Word.Length then
      Word := Copy(S, 1, pos(' ', S) - 1)
    Else
      Delete(S, 1, pos(' ', S));
    end;
    if length(S) > Word.Length then
      Word := S;
    end;
  F.Close;
  Writeln(Word);
end.

```

Python

```

f = open("D:/input.txt","r")
s = f.readlines()
f.close()
mx=0
word=""
for e in s:
    a=e.split(" ")[0]
    if len(a)>mx:
        word=a
        mx=len(a)
print(word)

```

Go

```

package main
import (
    "bufio"
    "fmt"
    "os"
    "strings"
)
func main() {
    f, _ := os.Open("input.txt")
    defer f.Close()
    var word string=" "
    scanner := bufio.NewScanner(f)
    for scanner.Scan() {
        s:= scanner.Text()
        t:=strings.Split(s, " ")[0]
        if len(t)>len(word){
            word=t
        }
    }
    fmt.Println(word)
}

```

8) Дан текстовый файл, содержащий текст, выровненный по левому краю. Выровнять текст по правому краю, добавив в начало каждой непустой строки нужное количество пробелов (ширину текста считать равной 50).

PascalABC.NET
<pre> Var F_in, F_out: Text; Name, S: string; begin Write('File name in: '); Readln(Name); Assign(F_in, Name); Assign(F_out, 'D:\temp.txt'); Reset(F_in); Rewrite(F_out); while not F_in.Eof do begin S := F_in.ReadlnString; while (S.Length <= 50) do S := ' ' + S; F_out.Writeln(F_out); end; F_in.Close; F_out.Close; F_in.Erase; F_out.Rename(Name); end. </pre>
Python
<pre> f = open("D:/input.txt","r") s = f.readlines() f.close() f = open("D:/input.txt","w") for e in s: if len(e)<=50: while len(e)<50: e=" "+e f.write(e) f.close() </pre>
Go
<pre> package main import ("bufio" "fmt" "os") func main() { f, _ := os.Open("input.txt") </pre>

```

fo, _ := os.Create("file.txt")
defer f.Close()
scanner := bufio.NewScanner(f)
for scanner.Scan() {
    s:= scanner.Text()
    if len(s)<=50{
        for len(s)<52{
            s=" "+s
        }
    }
    fmt.Fprintf(fo,"%s \n",s)
}
}

```

3. Текстовые файлы с числовой информацией

В заданиях на обработку текстовых файлов с числовой информацией предполагается, что изображения вещественных чисел, содержащиеся в текстовых файлах, удовлетворяют стандартным правилам используемого языка программирования (в частности, дробная часть отделяется от целой части десятичной *точкой*).

9) Даны два файла целых чисел одинакового размера. Создать текстовый файл, содержащий эти числа, расположенные в два столбца шириной по 30 символов (в первом столбце содержатся числа из первого исходного файла, во втором — из второго файла). В начало и конец каждой строки текстового файла добавить разделитель «|» (код 124). Числа выравниваются по правому краю столбца.

```

PascalABC.NET
Var
    F1_in, F2_in, F_out: Text;
    line, S, Name: string;
    i: integer;
begin
    Write('File 1 name in: ');
    ReadLn(Name);
    Assign(F1_in, Name);
    Write('File 2 name in: ');
    ReadLn(Name);
    Assign(F2_in, Name);
    Write('File name out: ');
    ReadLn(Name);
    Assign(F_out, Name);
    Reset(F1_in);
    Reset(F2_in);
    Rewrite(F_out);
    while (not eof(F1_in)) or (not eof(F2_in)) do
    begin
        line := '|';
        if (not eof(F1_in)) then
        begin
            read(F1_in, i);

```



```

    Str(i, S);
end
else S := '';
for i := 1 to 30 - Length(S) do
    line := line + ' ';
line := line + S;
if (not eof(F2_in)) then
begin
    read(F2_in, i);
    Str(i, S);
end
else S := '';
for i := 1 to 30 - Length(S) do
    line := line + ' ';
line := line + S;
line := line + '|';
Writeln(F_out, line);
end;
Close(F1_in);
Close(F2_in);
Close(F_out);
end.

```

Python

```

f = open("D:/input.txt","r")
s = f.readlines()
f.close()
x=[]
for e in s:
    a=e.split(" ")
    for i in range(0, len(a)-1):
        x.append(a[i])
print(len(x))
f = open("D:/input1.txt","r")
s = f.readlines()
f.close()
y=[]
for e in s:
    a=e.split(" ")
    for i in range(0, len(a)-1):
        y.append(a[i])
print(len(y))
f = open("D:/output.txt","w")
s=""
while len(s)<26:
    s=s+ " "
for i in range (0, len(x)-1):
    s0="|"+x[i]+s+"|"+y[i]+s+"|\n";
    f.write(s0)
f.close()

```

```
Go
```

```

package main
import (
    "bufio"
    "fmt"
    "os"
    "strings"
)
func main() {
    f1, _ := os.Open("input.txt")
    f2, _ := os.Open("input1.txt")
    fo, _ := os.Create("file.txt")
    defer f1.Close()
    defer f2.Close()
    scanner1 := bufio.NewScanner(f1)
    scanner2 := bufio.NewScanner(f2)
    for scanner1.Scan() && scanner2.Scan() {
        s1:=strings.Split(scanner1.Text()," ")
        s2:=strings.Split(scanner2.Text()," ")
        for i:=0; i<len(s2); i++){
            s:=" "
            for len(s)<30{
                s=s+" "
            }
            so:="|"+s1[i]+s+"|"+s2[i]+s+"|";
            fmt.Fprintf(fo,"%s \n", so)
        }
    }
}

```

10) Даны вещественные числа A , B и целое число N . Создать текстовый файл, содержащий таблицу значений функции \sqrt{x} на промежутке $[A, B]$ с шагом $(B - A)/N$. Таблица состоит из двух столбцов: с аргументами x (10 позиций, из них 4 под дробную часть) и со значениями \sqrt{x} (15 позиций, из них 8 под дробную часть). Столбцы выравниваются по правому краю.

PascalABC.NET	C++
<pre> Var F_out: Text; line, S, Name: string; A, B, N, i: integer; y, x: Real; begin Write('A: '); Readln(A); Write('B: '); Readln(B); Write('N: '); Readln(N); Write('File name out: '); </pre>	<pre> #include <iostream> #include <fstream> #include <string> #include <cmath> using namespace std; int main() { string file1 = "D:\input.txt"; ofstream ofs(file1); float x, y, a, b, n, h; string S; cin >> a >> b >> n; h = abs(b - a) / n; x = a; </pre>

<pre> Readln(Name); Assign(F_out, Name); Rewrite(F_out); y := A; x := 0; while x <= B do begin Write(F_out, ' '); str(trunc(x), S); for i:=1 to 5 - S.Length do F_out.Write(' '); Write(F_out, x:5:4); y := sqrt(x); str(trunc(y), S); for i:= 1 to 6 - S.Length do F_out.Write(' '); Write(F_out, y:6:8); F_out.Writeln(' '); x := x + (B - A) / N; end; F_out.Close; end. </pre>	<pre> S.append(6, ' '); while (x < b) { y = sqrt(x); ofs << " " << S << x << " " << S + ' ' << y << " " << endl; x+=h; } ofs.close(); system("pause"); return 0; } </pre>
Python	
<pre> import math f = open("D:/output.txt", "w") s="" while len(s)<6: s=s+" " a=float(input("a->")) b=float(input("b->")) n=int(input("n->")) h=abs(b-a)/n x=a while x<b: y=math.sqrt(x) s0=" "+str(x)+s+" "+str(y)+s+" \n" f.write(s0) print(s0) x+=h f.close() </pre>	
Go	
<pre> package main import ("fmt" "os" "math") func main() { var x,y,h,a,b,n float64 fmt.Scan(&a, &b,&n) h=math.Abs(b-a)/n fo, _ := os.Create("file.txt") defer fo.Close() </pre>	

```

for x=a; x<=b; x+=h{
    y=math.Sqrt(x)
    s=" "
    for len(s)<6{
        s=s+" "
    }
    so:= fmt.Sprintf("%f", x)+s
    s=" "
    for len(s)<6{
        s=s+" "
    }
    so= so+fmt.Sprintf("%f", y)+s
    fmt.Fprintf(fo,"%s \n", so)
    fmt.Println(so)
}
}

```

4. Дополнительные задания на обработку текстовых файлов

11) Дан текстовый файл. Создать символьный файл, содержащий все знаки препинания, встретившиеся в текстовом файле (в том же порядке).

```

PascalABC.NET
Var
    F_in: Text;
    F_out: file of Char;
    Name: string;
    C: Char;
begin
    Write('File In name: ');
    Readln(Name);
    Assign(F_in, Name);
    Reset(F_in);
    Write('File Out name: ');
    Readln(Name);
    Assign(F_out, Name);
    Rewrite(F_out);
    while not F_in.eof do
        begin
            while not F_in.eoln do
                begin
                    C := F_in.ReadChar;
                    if (char.IsPunctuation(C)) then
                        F_out.Write(C);
                end;
            F_in.Readln;
        end;
    F_in.Close;
    F_out.Close;
end.

```

Python

```
f = open("D:/input.txt","r+")
s=f.readlines()
f.close()
punctuations = '!()-[]{};:'\",<>./?@$%^&*~''''
f = open("D:/output.txt","w")
for e in s:
    for i in range(0, len(e)-1):
        if e[i] in punctuations:
            f.write(e[i])
```

Go

```
package main
import (
    "bufio"
    "fmt"
    "os"
    "unicode"
)
func main() {
    f, _ := os.Open("input.txt")
    fo, _ := os.Create("file.txt")
    defer f.Close()
    defer fo.Close()
    scanner := bufio.NewScanner(f)
    for scanner.Scan() {
        s := scanner.Text()
        for i:=0; i<len(s); i++){
            if unicode.IsPunct(rune(s[i])){
                fmt.Fprintf(fo,"%c ",rune(s[i]))
            }
        }
    }
}
```

Задачи для самостоятельной работы

Основные операции с текстовыми файлами

- 1) Дано имя файла и целое число N ($0 < N < 27$). Создать текстовый файл с указанным именем и записать в него N строк: первая строка должна содержать строчную (то есть маленькую) латинскую букву «а», вторая — буквы «ab», третья — буквы «abc» и т. д.; последняя строка должна содержать N начальных строчных латинских букв в алфавитном порядке.
- 2) Дано имя файла и целое число N ($0 < N < 27$). Создать текстовый файл с указанным именем и записать в него N строк длины N ; строка с номером K ($K = 1, \dots, N$) должна содержать K начальных *прописных* (то есть заглавных) латинских букв, дополненных справа символами «*» (звездочка). Например, для $N = 4$ файл должен содержать строки «A***», «AB**», «ABC*», «ABCD».
- 3) Дан текстовый файл. Вывести количество содержащихся в нем символов и строк (маркеры концов строк EOLN и конца файла EOF при подсчете количества символов не учитывать).
- 4) Даны два текстовых файла. Добавить в конец первого файла содержимое второго файла.
- 5) Дана строка S и текстовый файл. Добавить строку S в начало файла.
- 6) Даны два текстовых файла. Добавить в начало первого файла содержимое второго файла.
- 7) Дано целое число K и текстовый файл. Вставить пустую строку перед строкой файла с номером K . Если строки с таким номером нет, то оставить файл без изменений.
- 8) Дано целое число K и текстовый файл. Вставить пустую строку после строки файла с номером K . Если строки с таким номером нет, то оставить файл без изменений.
- 9) Дана строка S и текстовый файл. Заменить в файле все пустые строки на строку S .
- 10) Дан непустой текстовый файл. Удалить из него первую строку.
- 11) Дан непустой текстовый файл. Удалить из него последнюю строку.
- 12) Дано целое число K и текстовый файл. Удалить из файла строку с номером K . Если строки с таким номером нет, то оставить файл без изменений.
- 13) Дан текстовый файл. Удалить из него все пустые строки.
- 14) Дано целое число K и текстовый файл. Удалить из каждой строки файла первые K символов (если длина строки меньше K , то удалить из нее все символы).
- 15) Дан текстовый файл. Заменить в нем все прописные русские буквы на строчные, а все строчные — на прописные.
- 16) Дан текстовый файл. Заменить в нем все подряд идущие пробелы на один пробел.
- 17) Дан текстовый файл, содержащий более трех строк. Удалить из него последние три строки.
- 18) Дано целое число K ($0 < K < 10$) и текстовый файл, содержащий более K строк. Удалить из файла последние K строк.
- 19) Дано целое число K ($0 < K < 10$) и текстовый файл, содержащий более K строк. Создать новый текстовый файл, содержащий K последних строк исходного файла.

Анализ и форматирование текста

- 20) Дано целое число K и текстовый файл. Удалить из файла абзац с номером K (абзацы отделяются друг от друга одной или несколькими пустыми строками). Пустые строки, предшествующие и следующие за удаляемым абзацем, не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.
- 21) Дано целое число K и текстовый файл. Удалить из файла абзац с номером K . Пустые строки между абзацами не учитывать и не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.
- 22) Дан текстовый файл. Абзацы выделяются в нем с помощью *красной строки*, а пустых строк нет. Вставить между соседними абзацами по одной пустой строке (в начало и конец файла пустые строки не добавлять).
- 23) Дан текстовый файл. Вывести последнее слово текста наименьшей длины. *Словом* считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки.
- 24) Дано целое число K и текстовый файл. Создать строковый файл и записать в него все слова длины K из исходного файла. *Словом* считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки. Если исходный файл не содержит слов длины K , то оставить результирующий файл пустым.
- 25) Дан символ C — *прописная* (заглавная) русская буква и текстовый файл. Создать строковый файл и записать в него все слова из исходного файла, начинающиеся на эту букву (прописную или строчную). *Словом* считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки. Если исходный файл не содержит подходящих слов, то оставить результирующий файл пустым.
- 26) Дан текстовый файл, содержащий текст, выровненный по левому краю. Выровнять текст по центру, добавив в начало каждой непустой строки нужное количество пробелов (ширину текста считать равной 50). Строки нечетной длины перед центрированием дополнять слева пробелом.
- 27) Дан текстовый файл, содержащий текст, выровненный по правому краю. Выровнять текст по центру, удалив из каждой непустой строки половину начальных пробелов. В строках с нечетным количеством начальных пробелов перед центрированием удалять первый начальный пробел.
- 28) Дан текстовый файл, содержащий текст, выровненный по левому краю. Абзацы текста разделяются одной пустой строкой. Выровнять текст *по ширине* (то есть и по левому, и по правому краю), увеличив в каждой непустой строке (кроме последних строк абзацев) количество пробелов между словами, начиная с последнего пробела в строке (ширину текста считать равной 50).
- 29) Дано целое число K (> 25) и текстовый файл, содержащий текст, выровненный по левому краю. Абзацы текста отделяются друг от друга одной пустой строкой. Отформатировать текст так, чтобы его ширина не превосходила K позиций, и выровнять текст по левому краю, сохранив деление на абзацы. Пробелы в конце строк удалить. Сохранить отформатированный текст в новом текстовом файле.
- 30) Дано целое число K (> 25) и текстовый файл, содержащий текст, выровненный по левому краю. Абзацы выделяются в нем с помощью *красной строки* (5 начальных пробелов), а пустых строк нет. Отформатировать текст так, чтобы его

ширина не превосходила K позиций, и выровнять текст по левому краю, сохранив деление на абзацы. Пробелы в конце строк удалить. Сохранить отформатированный текст в новом текстовом файле.

Текстовые файлы с числовой информацией

- 31) Даны три файла целых чисел одинакового размера. Создать текстовый файл, содержащий эти числа, расположенные в три столбца шириной по 20 символов (в каждом столбце содержатся числа из соответствующего исходного файла). В начало и конец каждой строки текстового файла добавить разделитель «|» (код 124). Числа выравниваются по левому краю столбца.
- 32) Даны вещественные числа A , B и целое число N . Создать текстовый файл, содержащий таблицу значений функций $\sin(x)$ и $\cos(x)$ на промежутке $[A, B]$ с шагом $(B - A)/N$. Таблица состоит из трех столбцов: с аргументами x (8 позиций, из них 4 под дробную часть) и со значениями $\sin(x)$ и $\cos(x)$ (по 12 позиций, из них 8 под дробную часть). Столбцы выравниваются по правому краю.
- 33) Дан текстовый файл, каждая строка которого изображает целое число, дополненное слева и справа несколькими пробелами. Вывести количество этих чисел и их сумму.
- 34) Дан текстовый файл, каждая строка которого изображает целое или вещественное число, дополненное слева и справа несколькими пробелами (вещественные числа имеют ненулевую дробную часть). Вывести количество чисел с ненулевой дробной частью и их сумму.
- 35) Дан текстовый файл, каждая строка которого содержит изображения нескольких чисел, разделенные пробелами (вещественные числа имеют ненулевую дробную часть). Создать файл вещественных чисел, содержащий (в том же порядке) все числа из исходного файла, имеющие ненулевую дробную часть.
- 36) Дан текстовый файл, каждая строка которого изображает целое или вещественное число, дополненное слева и справа несколькими пробелами (вещественные числа имеют ненулевую дробную часть). Вывести количество целых чисел и их сумму.
- 37) Дан текстовый файл, каждая строка которого содержит изображения нескольких чисел, разделенные пробелами (вещественные числа имеют ненулевую дробную часть). Создать файл целых чисел, содержащий все целые числа из исходного файла (в том же порядке).
- 38) Дан текстовый файл и файл целых чисел. Добавить в конец каждой строки текстового файла изображение соответствующего числа из файла целых чисел. Если файл целых чисел короче текстового файла, то оставшиеся строки текстового файла не изменять.
- 39) Дан текстовый файл. В каждой его строке первые 30 позиций отводятся под текст, а оставшаяся часть — под вещественное число. Создать два файла: строковый файл, содержащий текстовую часть исходного файла, и файл вещественных чисел, содержащий числа из исходного файла (в том же порядке).
- 40) Дан текстовый файл, содержащий таблицу из трех столбцов вещественных чисел. Ширина столбцов таблицы и способ их выравнивания являются произвольными, специальных символов-разделителей таблица не содержит.

Создать три файла вещественных чисел, каждый из которых содержит числа из соответствующего столбца таблицы (в том же порядке).

- 41) Дан текстовый файл, содержащий таблицу из трех столбцов целых чисел. В начале и в конце каждой строки таблицы, а также между ее столбцами располагается *символ-разделитель*. Ширина столбцов таблицы, способ их выравнивания и вид символа-разделителя являются произвольными. Создать файл целых чисел, содержащий сумму чисел из каждой строки исходной таблицы.

Дополнительные задания на обработку текстовых файлов

- 42) Дан текстовый файл. Создать символьный файл, содержащий все символы, встретившиеся в тексте, включая пробел и знаки препинания (без повторений). Символы располагать в порядке их первого появления в тексте.
- 43) Дан текстовый файл. Создать символьный файл, содержащий все символы, встретившиеся в тексте, включая пробел и знаки препинания (без повторений). Символы располагать в порядке возрастания их кодов.
- 44) Дан текстовый файл. Создать символьный файл, содержащий все символы, встретившиеся в тексте, включая пробел и знаки препинания (без повторений). Символы располагать в порядке убывания их кодов.
- 45) Дан текстовый файл. Подсчитать число появлений в нем каждой *строчной* (то есть маленькой) русской буквы и создать строковый файл, элементы которого имеют вид «<буква>—<число ее появлений>» (например, «а–25»). Буквы, отсутствующие в тексте, в файл не включать. Строки упорядочить по возрастанию кодов букв.
- 46) Дан текстовый файл. Подсчитать число появлений в нем каждой *строчной* (то есть маленькой) русской буквы и создать строковый файл, элементы которого имеют вид «<буква>—<число ее появлений>» (например, «а–25»). Буквы, отсутствующие в тексте, в файл не включать. Строки упорядочить по убыванию числа появлений букв, а при равном числе появлений — по возрастанию кодов букв.
- 47) Дана строка S , состоящая из 10 цифр, и файл с русским текстом. Зашифровать файл, выполнив циклическую замену каждой русской буквы, стоящей на K -й позиции строки, на букву того же регистра, расположенную в алфавите на S_K -м месте после шифруемой буквы (для $K = 11$ снова используется смещение S_1 и т. д.). Букву «ё» в алфавите не учитывать, знаки препинания и пробелы не изменять.
- 48) Дана строка и файл с русским текстом, зашифрованным по правилу, описанному в задании Text59. Данная строка представляет собой первую расшифрованную строку текста. Расшифровать остальные строки и заменить в файле зашифрованный текст на расшифрованный. Если информации для расшифровки недостаточно, то исходный файл не изменять.

16. Составные типы данных в процедурах и функциях: группа Pascal

В каждом задании данного раздела требуется описать процедуру или функцию и затем использовать ее для обработки исходных данных.

Все параметры любой *функции* считаются входными. Для *процедур* всегда указывается, какие параметры являются выходными (или одновременно входными и выходными); если о виде параметра процедуры ничего не сказано, то он считается входным.

1. Одномерные и двумерные массивы

При вводе исходного массива вначале следует ввести его размер (одно число для одномерных массивов, два числа — количество строк и столбцов — для двумерных массивов-матриц), а затем — все его элементы.

Если в задании явно не указывается размер одномерного массива, являющегося параметром процедуры или функции, то предполагается, что этот размер может меняться в пределах от 1 до 10. Для двумерных массивов-матриц также предполагается, что число их строк и столбцов может меняться от 1 до 10. Порядковые номера начальных элементов как одномерных, так и двумерных массивов всегда считаются равными 1.

При описании процедур, выполняющих преобразование массива, не следует использовать вспомогательный массив того же размера

- 1) Описать функцию MinInt(A, N) целого типа, находящую минимальный элемент целочисленного массива A размера N. С помощью этой функции найти минимальные элементы массивов A, B, C размера N_A, N_B, N_C соответственно.

PascalABC.NET	C++
<pre> Var A,B,C: array of integer; Na, Nb, Nc: integer; function MinInt(a: array of integer; N: integer): Integer; var MinEl, i: integer; begin MinEl := 1; for i := 1 to N do if (A[i]<A[MinEl]) then MinEl := i; MinInt := MinEl; end; procedure EnterArray(a: array of integer; var N: integer); var i: integer; begin for i := 1 to N do A[i] := random(50); end; </pre>	<pre> #include <iostream> using namespace std; int minInt(int x[], int n) { int mn = x[0]; for (int i = 1; i < n; i++) { if (x[i] < mn) mn = x[i]; } return mn; } int *init (int n) { int *x = new int[n]; for (int i = 0; i < n; i++) { x[i] = rand() % 50; } return x; } void printArray(int a[], int n) { for (int i = 0; i < n; i++) cout << a[i] << " "; } int main(){ int *a,*b,*c,Na, Nb, Nc, i; cin >> Na >> Nb >> Nc; a = init(Na); </pre>

<pre> procedure PrintArray(a: array of integer; var N: integer); var i: integer; begin for i := 1 to N do print(a[i]); writeln; end; begin Writeln('Enter N:'); Na := ReadInteger; a := new integer[Na + 1]; EnterArray(A, Na); PrintArray(A, Na); Writeln(MinInt(A, Na)); Writeln('Enter N:'); Nb := ReadInteger; B := new integer[Nb + 1]; EnterArray(B, Nb); PrintArray(B, Nb); Writeln(MinInt(B, Nb)); Writeln('Enter N:'); Nc := ReadInteger; C := new integer[Nc + 1]; EnterArray(C, Nc); PrintArray(C, Nc); Writeln(MinInt(C, Nc)); end. </pre>	<pre> b = init(Nb); c = init(Nc); printArray(a, Na); cout<<"\n"<<minInt(a,Na)<< endl; printArray(b, Nb); cout<<"\n"<<minInt(b,Nb)<<endl; printArray(c, Nc); cout<<"\n"<<minInt(c,Nc)<<endl; system("pause"); return 0; } </pre>
Python	C#
<pre> def minInt(a,n): a=[] for i in range (0, n): a.append(int(input())) return min(a) n=int(input("n->")) a=[] print (minInt(a,n)) </pre>	<pre> static int MinInt(int[] x){ int mn = x[0]; for (int i = 1; i < x.Length; i++) if (mn > x[i]) mn = x[i]; return mn; } static int[] init(int n){ int[] x = new int[n]; Random rnd = new Random(); for (int i = 0; i < n; i++) x[i] = rnd.Next(n, n*10); PrintArray(x); return x; } static void PrintArray(int[]x){ for (int i = 0; i < x.Length; i++) Console.Write(x[i] + " "); Console.WriteLine(); } static void Main(string[] args) { </pre>

	<pre> int Na, Nb, Nc; Na = Convert.ToInt32(Console.ReadLine()); Nb = Convert.ToInt32(Console.ReadLine()); Nc= Convert.ToInt32(Console.ReadLine()); int[] a = init(Na); int[] b = init(Nb); int[] c = init(Nc); Console.WriteLine(MinInt(a) + " " + MinInt(b) + " " + MinInt(c)); Console.ReadKey(); } </pre>
Java	Go
<pre> static int MinInt(int[]x){ int mn=x[0]; for (int i=1; i<x.length; i++) if (mn>x[i]) mn=x[i]; return mn; } static int []init(int n){ int []x=new int[n]; Random rnd=new Random(); for (int i=0; i<n; i++) x[i]=rnd.nextInt(30); PrintArray(x); return x; } static void PrintArray(int []x){ for (int i=0; i<x.length; i++) System.out.print(x[i]+" "); System.out.println(); } public static void main(String[] args){ int Na,Nb,Nc; Scanner sc=new Scanner(System.in); Na=sc.nextInt(); Nb=sc.nextInt(); Nc=sc.nextInt(); int []a=init(Na); int []b=init(Nb); int []c=init(Nc); System.out.println(MinInt(a)+" "+MinInt(b)+" "+MinInt(c)); } </pre>	<pre> package main import ("fmt" "math/rand") func MinInt(a[] byte)byte{ var min=a[0] for i:=1; i<len(a); i++){ if min>a[i]{ min=a[i] } } return min } func main() { var Na,Nb, Nc int fmt.Println("Na, Nb, Nc->") fmt.Scan(&Na, &Nb, &Nc) a := make([]byte, Na) rand.Read(a) fmt.Println(a) fmt.Println(MinInt(a)) b := make([]byte, Nb) rand.Read(b) fmt.Println(b) fmt.Println(MinInt(b)) c := make([]byte, Nc) rand.Read(c) fmt.Println(c) fmt.Println(MinInt(c)) } </pre>
Kotlin	
<pre> import java.util.Scanner fun IntMin(x: Array<Int>, n:Int):Int{ var mn=x[0]; for (i in (1..n-1)){ </pre>	

<pre> if (x[i]<mn) mn=x[i] } return mn } fun main(args: Array<String>) { val reader = Scanner(System.`in`) val n = reader.nextInt(); val x = Array(n,{i->2*i+1}) for(i in 0..n-1) { println(x[i]) } print(IntMin(x,n)) } </pre>
VB.NET
<pre> Function minInt(x() As Integer, n As Integer) As Integer Dim min As Integer = x(0) For i As Integer = 1 To n - 1 If min > x(i) Then min = x(i) End If Next Return min End Function Sub Main() Dim i, n, min As Integer n = Convert.ToInt32(Console.ReadLine()) Dim a() As Integer = New Integer(n) {} Dim rnd As Random = New Random() For i = 0 To n - 1 a(i) = rnd.Next(50) Console.Write("{0} ", a(i)) Next Console.WriteLine() min = minInt(a, n) Console.WriteLine(min) Console.ReadKey() End Sub </pre>

2) Описать функцию $\text{Norm1}(A, M, N)$ вещественного типа, вычисляющую *норму* вещественной матрицы A размера $M \times N$:

$$\text{Norm1}(A, M, N) = \max \{|A_{1,j}| + |A_{2,j}| + \dots + |A_{M,j}|\},$$

где максимум берется по всем J от 1 до N . Для данной матрицы A размера $M \times N$ найти $\text{Norm1}(A, K, N)$, $K = 1, \dots, M$.

PascalABC.NET	C++
<pre> Var M, N, K: integer; A: array[,] of integer; function Norm1(var A: array[,] of integer; M: integer; N: integer): Real; begin var NormMax := 0; </pre>	<pre> #include <iostream> using namespace std; void fill_array(int **arr, int m, int n) { int count = 0; for (int i = 0; i<m; i++) { for (int j = 0; j<n; j++) { arr[i][j] = rand()%30; } } } </pre>

<pre> for var i := 1 to M do begin var Norm := 0; for var j := 1 to N do Norm := Norm + a[i, j]; if (Norm>NormMax) or (i=1) then NormMax := Norm; end; Norm1 := NormMax; end; procedure InitMatrix(var B: array[,] of integer; var M, N: integer); begin for var i := 1 to M do begin for var j := 1 to N do B[i, j] := random(50); end; Writeln; end; end; procedure PrintMatrix(var B: array[,] of integer; var M, N: integer); begin for var i := 1 to M do begin for var j := 1 to N do print(B[i, j]); Writeln; end; Writeln; end; begin Write('M: '); Readln(M); Write('N: '); Readln(N); A:=new integer[M + 1, N + 1]; InitMatrix(A, M, N); PrintMatrix(A, M, N); for K := 1 to M do Writeln(Norm1(A, K, N)); end. </pre>	<pre> } } void print_array(int** arr, int m, int n) { for (int i = 0; i < m; i++) { for (int j = 0; j < n; j++) cout << arr[i][j] << '\t'; cout << '\n'; } cout << '\n'; } int Norm1(int **x, int m, int n) { int NormMax = 0; for (int i = 0; i < m; i++) { int Norm = 0; for (int j = 0; j < n; j++) { Norm = Norm + x[i][j]; if ((Norm>NormMax) (i == 0)) NormMax = Norm; } } return NormMax; } int main() { int m, n; cout << "m, n: "; cin >> m >> n; int **x = new int*[m]; for (int i = 0; i<m; i++) { x[i] = new int[n]; } fill_array(x, m, n); print_array(x, m, n); cout << Norm1(x, m, n) << endl; for (int i = 0; i<m; i++) { delete[]x[i]; } delete[]x; system("pause"); return 0; } </pre>
Python	C#
<pre> import random def init(c, m, n): for i in range(m): for j in range(n): c[i][j] = random.randrange(0,101, 5) </pre>	<pre> static int Norm1(int[,]x, int m, int n){ int NormMax=0; for (int i = 0; i < m; i++) { int Norm=0; for (int j = 0; j < n; j++) { Norm = Norm + x[i,j]; </pre>

<pre> return c def Norm1(c, m, n): NormMax=0 for i in range(m): Norm=0 for j in range(m): Norm=Norm+c[i][j]; if Norm>NormMax or i==0: NormMax = Norm return NormMax def show(a, m, n): init(a, 5, 5) for i in range(m): for j in range(m): print(a[i][j], end=' ') print() a = [[2]*5 for _ in range(5)] show(a, 5,5) print(Norm1(a,5,5)) </pre>	<pre> if ((Norm>NormMax) (i==0)) NormMax = Norm; } } return NormMax; } static int [,]init(int m, int n){ int [,]x=new int[m,n]; Random rnd=new Random(); for (int i=0; i<m; i++) for (int j=0; j<n; j++) x[i,j]=rnd.Next(50); PrintArray(x, m, n); return x; } static void PrintArray(int [,]x, int m, int n){ for (int i=0; i<m; i++){ for (int j=0; j<n; j++) Console.Write(x[i,j]+" "); Console.WriteLine(); } Console.WriteLine(); } static void Main(string[] args) { int m, n; m=Convert.ToInt32(Console.ReadLine()); n=Convert.ToInt32(Console.ReadLine()); int[,]a=init(m,n); Console.WriteLine(Norm1(a, m, n)); Console.ReadKey(); } </pre>
Java	Go
<pre> static int Norm1(int[][]x,int m,int n){ int NormMax=0; for (int i = 0; i < m; i++) { int Norm=0; for (int j = 0; j < n; j++) { Norm = Norm + x[i][j]; if ((Norm>NormMax) (i==0)) NormMax = Norm; } } return NormMax; } static int [][]init(int m, int n){ int [][]x=new int[m][n]; Random rnd=new Random(); for (int i=0; i<m; i++) for (int j=0; j<n; j++) x[i][j]=rnd.nextInt(30); PrintArray(x, m, n); } </pre>	<pre> package main import ("fmt" "math/rand") func print2D(arr [][]int) { for i:=0;i<len(arr); i++ { forj:=0;j<len(arr[0]);j++ { fmt.Printf("%d ",arr[i][j]) } fmt.Println() } } func Norm1(arr [][]int) int{ NormMax:=0 for i:=0; i<len(arr);i++){ Norm:=0 for j:=0;j<len(arr[0]);j++ { </pre>

<pre> return x; } static void PrintArray(int [][]x, int m, int n){ for (int i=0; i<m; i++){ for (int j=0; j<n; j++) System.out.print(x[i][j]+" "); System.out.println(); } System.out.println(); } public static void main(String[] args){ int m,n; Scanner sc=new Scanner(System.in); m=sc.nextInt(); n=sc.nextInt(); int[][]a=init(m,n); System.out.println(Norm1(a, m, n)); } </pre>	<pre> Norm = Norm + arr[i][j]; if Norm>NormMax i==0{ NormMax = Norm } } return NormMax } func main() { var arr [][]int var m int fmt.Println("m->") fmt.Scan(&m) for i := 0; i < m; i++ { tmp := make([]int, m) for j := 0; j < m; j++ { tmp[j]=rand.Intn(50) } arr = append(arr, tmp) } print2D(arr) fmt.Println(Norm1(arr)) } </pre>
---	---

2. Строки

- 3) Описать функцию IsIdent(S) целого типа, проверяющую, является ли строка S допустимым *идентификатором*, то есть непустой строкой, которая содержит только латинские буквы, цифры и символ подчеркивания « $_$ » и не начинается с цифры. Если S является допустимым идентификатором, то функция возвращает 0. Если S является пустой строкой, то возвращается -1 , если S начинается с цифры, то возвращается -2 . Если S содержит недопустимые символы, то возвращается номер первого недопустимого символа. Проверить с помощью функции IsIdent пять данных строк.

<p>PascalABC.NET</p> <pre> Var S: String; i: integer; function IsIdent(S: String): integer; var i, temp: integer; begin temp := 0; if length(s) > 0 then if (s[1] in ['a'..'z', '_']) then begin for i := length(S) downto 2 do if not (s[i] in ['0'..'9', 'a'..'z', '_']) then temp := i; </pre>
--

<pre> end else temp := -2 else temp := -1; IsIdent := temp; end; begin for i := 1 to 5 do begin Write('S:'); Readln(S); Writeln(IsIdent(S)); end; end. </pre>
Python
<pre> def IsIndent(S): temp=0 if len(S)>0: if S[0] in [range(ord('a'), ord('z')), '_']: for i in range (len(S), 1, -1): if S[i] not in [range(ord('a'), ord('z')), range(0,9), '_']: temp=i else: temp=-2 else: temp=-1 return temp for i in range (0, 5): S=input("S->") print(IsIndent(S)) </pre>

- 4) Описать функцию CompressStr(S) строкового типа, выполняющую *сжатие* строки S по следующему правилу: каждая подстрока строки S, состоящая из более чем четырех одинаковых символов C, заменяется текстом вида «C{K}», где K — количество символов C (предполагается, что строка S не содержит фигурных скобок «{» и «}»). Например, для строки S = «bbbсссссе» функция вернет строку «bbbc{5}e». С помощью функции CompressStr сжать пять данных строк.

PascalABC.NET
<pre> Var S: String; i: integer; function CompressStr(S: String): String; var i, i2, Num: integer; Stemp: String; </pre>

```

    C: Char;
begin
    C := S[1];
    Num := 1;
    Result := '';
    for i := 2 to length(S) do
    begin
        if S[i] = C then inc(num);
        if (S[i] <> C) or (i = length(S)) then
        begin
            if num > 3 then
            begin
                Str(num, Stemp);
                Result := Result + C + '{' + Stemp + '}';
            End
            else
                for i2 := 1 to num do
                    Result := Result + C;
                C := S[i];
                Num := 1;
            end;
        end;
    end;
end;

begin
    for i := 1 to 5 do
    begin
        Write('S:');
        Readln(S);
        Writeln(CompressStr(S));
    end;
end.

```

Python

```

def CompressStr(S):
    C=S[0]
    Num=1
    Rezult=""
    for i in range (1, len(S)):
        if S[i]==C:
            Num+=1
        if S[i]!=C or i==len(S)-1:
            if Num>3:
                Rezult=Rezult+C+"{"+str(Num)+"}"
            else:
                for j in range (0, Num):
                    Rezult = Rezult + C;
            C=S[i]
            Num=1
    return Rezult

```

```

for i in range (0, 5):
    S=input("S->")
    print(CompressStr(S))

```

Go

```

package main
import (
    "fmt"
    "strconv"
)
func CompressStr(S string) string{
C:=S[0]
Num:=1
Rezult:=""
for i:=1; i<len(S); i++){
    if S[i]==C{
        Num+=1
    }
    if S[i]!=C || i==len(S)-1{
        if Num>3{
            Rezult=Rezult+string(C)+"{"+strconv.Itoa(Num)+"}"
        }else{
            for j:=0; j<Num; j++){
                Rezult = Rezult + string(C)
            }
        }
        C=S[i]
        Num=1
    }
}
return Rezult
}
func main() {
    var S string
    fmt.Scan(&S)
    fmt.Println(CompressStr(S))
}

```

3. Файлы

5) Описать процедуру $AddLineNumbers(S, N, K, L)$, добавляющую в начало каждой строки существующего текстового файла с именем S ее порядковый номер: первая строка получает номер N , вторая $N + 1$ и т. д. Номер отображается в K позициях, выравнивается по правому краю и отделяется от последующего текста L пробелами ($K > 0, L > 0$). Если строка файла является пустой, то она также нумеруется, но пробелы после номера не добавляются. Применить эту процедуру к данному файлу, используя указанные значения N, K и L .

PascalABC.NET

Type

```

    FileName = String;
var
    N, K, L: integer;
    S: FileName;
procedure AddLineNumbers(S: FileName; N, K, L: integer);
var
    F, F_temp: Text;
    i: integer;
    TempS, Line: String;
begin
    Assign(F, S);
    Reset(F);
    Assign(F_temp, 'temp.txt');
    Rewrite(F_temp);
    while (not F.eof) do
    begin
        TempS := '';
        str(N, TempS);
        for i := TempS.Length to K - 1 do
            TempS := ' ' + TempS;
        Line := F.ReadLnString;
        if Line.Length > 0 then
            for i := 1 to L do
                TempS := TempS + ' ';
            TempS := TempS + Line;
            F_temp.WriteLine(TempS);
            inc(N);
        end;
        F.Close;
        F_temp.Close;
        F.Erase;
        F_temp.Rename(S);
    end;

begin
    Write('S: ');
    ReadLn(S);
    Write('N: ');
    ReadLn(N);
    Write('K: ');
    ReadLn(K);
    Write('L: ');
    ReadLn(L);
    AddLineNumbers(S, N, K, L);
end.

```

Python

```

def FileRead(S,n,k,l):
    f=open(S,"r")
    s=f.readlines()
    f.close()
    S0=""
    S1=""
    for i in range(0, k):
        S0=S0+" "
    for i in range(0, l):
        S1=S1+" "
    num=n
    f=open(S,"w")
    for e in s:
        if len(e)<=1:
            SS=S0+str(num)
        else:
            SS=S0+str(num)+S1+e
        num+=1
        f.write(SS)
n=int(input("n->"))
k=int(input("k->"))
l=int(input("l->"))
FileRead("D:/input.txt",n,k,l);

```

Go

```

package main
import (
    "bufio"
    "fmt"
    "log"
    "os"
)
func FileRead(S string,n,k,l int){
    file, err := os.Open(S)
    if err != nil {
        log.Fatal(err)
    }
    S0:= ""
    S1:= ""
    for i:=0;i<k; i++){
        S0=S0+" "
    }
    for i:=0; i<l; i++){
        S1=S1+" "
    }
    num:=n
    fileo, err1:= os.Create("D:/temp.txt")
    if err1 != nil{
        fmt.Println(err1)
        os.Exit(1)
    }
}

```

```

    }
    scanner := bufio.NewScanner(file)
    for scanner.Scan() {
        var SS string=""
        e:=scanner.Text()
        if len(e)<=1{
            SS=S0+string(num)
        }else{
            SS=S0+string(num)+S1+e
        }
        num+=1
        fmt.Fprintf( fileo,"%s \n", SS)
    }
    file.Close()
    fileo.Close()
    os.Rename("D:/temp.txt",S)
}
func main() {
    var n,k,l int
    fmt.Scan(&n,&k,&l)
    FileRead("D:/input.txt",n,k,l);
}

```

4. Записи

При вводе и выводе каждой даты вначале указывается день, затем номер месяца, затем год. При вводе каждой точки вначале указывается ее абсцисса (x-координата), затем ее ордината (y-координата).

- б) Описать тип TDate — запись с полями целого типа Day (день), Month (месяц) и Year (год) — и функцию LeapYear(D) логического типа с параметром типа TDate, которая возвращает True, если год в дате D является високосным, и False в противном случае. Вывести значение функции LeapYear для пяти данных дат (предполагается, что все даты являются правильными). *Високосным* считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400.

PascalABC.NET	C++
Type TDate = record Day: integer; Month: integer; Year: integer; end; var D: TDate; function Leap_Year(D: TDate): boolean; begin if (D.Year mod 100) = 0 then	struct TDate { public: int Day, Month, Year; }; bool LeaP_Year(TDate D) { if (D.Year % 100 == 0) return D.Year % 400 == 0; else return D.Year % 4 == 0; } int main(){ TDate d; cin>>d.Day>>d.Month>>d.Year; cout << LeaP_Year(d); system("pause"); return 0; }

<pre> if (D.Year mod 400) = 0 then LeaP_Year := True else LeaP_Year := False else if (D.Year mod 4) = 0 then LeaP_Year := True else LeaP_Year := False; end; begin Write('Date: '); Readln(D.Day, D.Month, D.Year); Writeln(LeaP_Year(D)); end. </pre>	
<p style="text-align: center;">Python</p>	<p style="text-align: center;">C#</p>
<pre> class TDate(object): def __init__(self, Day, Month, Year): self.Day = Day self.Month = Month self.Year = Year def LeaP_Year(D: TDate): if D.Year % 100 == 0: if D.Year % 400 ==0: return True else: return False else: if D.Year % 4 == 0: return True else: return False x=TDate(int(input()), int(input()), int(input())) print(LeaP_Year(x)) </pre>	<pre> struct TDate{ public int Day, Month, Year; }; static bool LeaP_Year(TDate D){ if (D.Year % 100 == 0) return D.Year % 400 == 0; else return D.Year % 4 == 0; } static void Main(string[] args) { TDate d; d.Day=Convert.ToInt32(Console.ReadLine()); d.Month=Convert.ToInt32(Console.ReadLine()); d.Year=Convert.ToInt32(Console.ReadLine()); Console.WriteLine(LeaP_Year(d)); Console.ReadKey(); } </pre>
<p style="text-align: center;">Java</p>	<p style="text-align: center;">Go</p>
<pre> static boolean LeaP_Year(TDate D){ if (D.Year % 100 == 0){ return D.Year % 400 ==0; } else return D.Year % 4 == 0; } public static void main(String[] args) { TDate d=new TDate(); </pre>	<pre> package main import "fmt" type TData struct { Day int Month int Year int } func LeaP_Year(D TData) bool{ if D.Year % 100 == 0{ if D.Year % 400 ==0{ </pre>

```
Scanner sc=new Scanner(System.in);
d.Day=sc.nextInt();
d.Month=sc.nextInt();
d.Year=sc.nextInt();
System.out.println(LeaP_Year(d));
}
```

```
        return true
    }else{
        return false
    }
}else{
    if D.Year % 4 == 0{
        return true
    } else{
        return false
    }
}
}

func main() {
    var d = TData{25,4, 2000}
    fmt.Println(LeaP_Year(d))
}
```


Задачи для самостоятельной работы

В каждом задании данного раздела требуется описать процедуру или функцию и затем использовать ее для обработки исходных данных.

Все параметры любой *функции* считаются входными. Для *процедур* всегда указывается, какие параметры являются выходными (или одновременно входными и выходными); если о виде параметра процедуры ничего не сказано, то он считается входным.

Одномерные и двумерные массивы

- 1) Описать функцию $NMax(A, N)$ целого типа, находящую номер максимального элемента вещественного массива A размера N . С помощью этой функции найти номера максимальных элементов массивов A, B, C размера N_A, N_B, N_C соответственно.
- 2) Описать процедуру $NMinMax(A, N, NMin, NMax)$, находящую номера минимального и максимального элемента вещественного массива A размера N . Выходные параметры целого типа: $NMin$ (номер минимального элемента) и $NMax$ (номер максимального элемента). С помощью этой процедуры найти номера минимальных и максимальных элементов массивов A, B, C размера N_A, N_B, N_C соответственно.
- 3) Описать процедуру $Invert(A, N)$, меняющую порядок следования элементов вещественного массива A размера N на противоположный (*инвертирование массива*). Массив A является входным и выходным параметром. С помощью этой процедуры инвертировать массивы A, B, C размера N_A, N_B, N_C соответственно.
- 4) Описать процедуру $Smooth1(A, N)$, выполняющую *сглаживание* вещественного массива A размера N следующим образом: элемент A_K заменяется на среднее арифметическое первых K исходных элементов массива A . Массив A является входным и выходным параметром. С помощью этой процедуры выполнить пятикратное сглаживание данного массива A размера N , выводя результаты каждого сглаживания.
- 5) Описать процедуру $Smooth2(A, N)$, выполняющую *сглаживание* вещественного массива A размера N следующим образом: элемент A_1 не изменяется, элемент A_K ($K = 2, \dots, N$) заменяется на полусумму исходных элементов A_{K-1} и A_K . Массив A является входным и выходным параметром. С помощью этой процедуры выполнить пятикратное сглаживание данного массива A размера N , выводя результаты каждого сглаживания.
- 6) Описать процедуру $Smooth3(A, N)$, выполняющую *сглаживание* вещественного массива A размера N следующим образом: каждый элемент массива заменяется на его среднее арифметическое с соседними элементами (при вычислении среднего арифметического используются *исходные* значения соседних элементов). Массив A является входным и выходным параметром. С помощью этой процедуры выполнить пятикратное сглаживание данного массива A размера N , выводя результаты каждого сглаживания.
- 7) Описать процедуру $RemoveX(A, N, X)$, удаляющую из целочисленного массива A размера N элементы, равные целому числу X . Массив A и число N являются входными и выходными параметрами. С помощью этой процедуры удалить числа X_A, X_B, X_C из массивов A, B, C размера N_A, N_B, N_C соответственно и вывести размер и содержимое полученных массивов.

- 8) Описать процедуру $\text{RemoveForInc}(A, N)$, удаляющую из вещественного массива A размера N «лишние» элементы так, чтобы оставшиеся элементы оказались упорядоченными по возрастанию: первый элемент не удаляется, второй элемент удаляется, если он меньше первого, третий — если он меньше предыдущего элемента, оставленного в массиве, и т. д. Например, массив 5.5, 2.5, 4.6, 7.2, 5.8, 9.4 должен быть преобразован к виду 5.5, 7.2, 9.4. Массив A и число N являются входными и выходными параметрами. С помощью этой процедуры преобразовать массивы A, B, C размера N_A, N_B, N_C соответственно и вывести размер и содержимое полученных массивов.
- 9) Описать процедуру $\text{DoubleX}(A, N, X)$, дублирующую в целочисленном массиве A размера N элементы, равные целому числу X . Массив A и число N являются входными и выходными параметрами. С помощью этой процедуры продублировать числа X_A, X_B, X_C в массивах A, B, C размера N_A, N_B, N_C соответственно и вывести размер и содержимое полученных массивов.
- 10) Описать процедуру $\text{Sort}(A, N)$, выполняющую сортировку по возрастанию вещественного массива A размера N . Массив A является входным и выходным параметром. С помощью этой процедуры отсортировать массивы A, B, C размера N_A, N_B, N_C соответственно.
- 11) Описать процедуру $\text{SortIndex}(A, N, I)$, формирующую для вещественного массива A размера N индексный массив I — массив целых чисел того же размера, содержащий номера элементов массива A в том порядке, который соответствует возрастанию элементов массива A (сам массив A при этом не изменяется). Индексный массив I является выходным параметром. С помощью этой процедуры создать индексные массивы для массивов A, B, C размера N_A, N_B, N_C соответственно.
- 12) Описать процедуру $\text{Bell}(A, N)$, меняющую порядок элементов вещественного массива A размера N на следующий: наименьший элемент массива располагается на первом месте, наименьший из оставшихся элементов — на последнем, следующий по величине располагается на втором месте, следующий — на предпоследнем и т. д. (в результате график значений элементов будет напоминать колокол). Массив A является входным и выходным параметром. С помощью этой процедуры преобразовать массивы A, B, C размера N_A, N_B, N_C соответственно.
- 13) Описать процедуру $\text{Split2}(A, N_A, B, N_B, C, N_C)$, формирующую по целочисленному массиву A размера N_A два целочисленных массива B и C размера N_B и N_C соответственно; при этом массив B содержит все четные числа из массива A , а массив C — все нечетные числа (в том же порядке). Массивы B и C и числа N_B и N_C являются выходными параметрами. Применить эту процедуру к данному массиву A размера N_A и вывести размер и содержимое полученных массивов B и C .
- 14) Описать процедуру $\text{ArrayToMatrix1}(A, K, M, N, B)$, формирующую по вещественному массиву A размера K матрицу B размера $M \times N$ (матрица заполняется элементами массива A по строкам). «Лишние» элементы массива игнорируются; если элементов массива недостаточно, то оставшиеся элементы матрицы полагаются равными 0. Двумерный массив B является выходным параметром. С помощью этой процедуры на основе данного массива A размера K и целых чисел M и N сформировать матрицу B размера $M \times N$.

- 15) Описать процедуру $\text{ArrayToMatrix2}(A, K, M, N, B)$, формирующую по вещественному массиву A размера K матрицу B размера $M \times N$ (матрица заполняется элементами массива A по столбцам). «Лишние» элементы массива игнорируются; если элементов массива недостаточно, то оставшиеся элементы матрицы полагаются равными 0. Двумерный массив B является выходным параметром. С помощью этой процедуры на основе данного массива A размера K и целых чисел M и N сформировать матрицу B размера $M \times N$.
- 16) Описать процедуру $\text{Chessboard}(M, N, A)$, формирующую по целым положительным числам M и N матрицу A размера $M \times N$, которая содержит числа 0 и 1, расположенные в «шахматном» порядке, причем $A_{1,1} = 0$. Двумерный целочисленный массив A является выходным параметром. С помощью этой процедуры по данным целым числам M и N сформировать матрицу A размера $M \times N$.
- 17) Описать функцию $\text{Norm2}(A, M, N)$ вещественного типа, вычисляющую *норму* вещественной матрицы A размера $M \times N$:
- $$\text{Norm2}(A, M, N) = \max \{|A_{I,1}| + |A_{I,2}| + \dots + |A_{I,N}|\},$$
- где максимум берется по всем I от 1 до M . Для данной матрицы A размера $M \times N$ найти $\text{Norm2}(A, K, N)$, $K = 1, \dots, M$.
- 18) Описать функцию $\text{SumRow}(A, M, N, K)$ вещественного типа, вычисляющую сумму элементов вещественной матрицы A размера $M \times N$, расположенных в K -й строке (если $K > M$, то функция возвращает 0). Для данной матрицы A размера $M \times N$ и трех данных K найти $\text{SumRow}(A, M, N, K)$.
- 19) Описать функцию $\text{SumCol}(A, M, N, K)$ вещественного типа, вычисляющую сумму элементов вещественной матрицы A размера $M \times N$, расположенных в K -м столбце (если $K > N$, то функция возвращает 0). Для данной матрицы A размера $M \times N$ и трех данных K найти $\text{SumCol}(A, M, N, K)$.
- 20) Описать процедуру $\text{SwapRow}(A, M, N, K_1, K_2)$, осуществляющую перемену местами строк вещественной матрицы A размера $M \times N$ с номерами K_1 и K_2 . Матрица A является входным и выходным параметром; если K_1 или K_2 больше M , то матрица не изменяется. Используя эту процедуру, поменять для данной матрицы A размера $M \times N$ строки с данными номерами K_1 и K_2 .
- 21) Описать процедуру $\text{SwapCol}(A, M, N, K_1, K_2)$, осуществляющую перемену местами столбцов вещественной матрицы A размера $M \times N$ с номерами K_1 и K_2 . Матрица A является входным и выходным параметром; если K_1 или K_2 больше N , то матрица не изменяется. Используя эту процедуру, поменять для данной матрицы A размера $M \times N$ столбцы с данными номерами K_1 и K_2 .
- 22) Описать процедуру $\text{Transp}(A, M)$, выполняющую *транспонирование* (то есть зеркальное отражение относительно главной диагонали) квадратной вещественной матрицы A порядка M . Матрица A является входным и выходным параметром. Используя эту процедуру, транспонировать данную матрицу A порядка M .
- 23) Описать процедуру $\text{RemoveRows}(A, M, N, K_1, K_2)$, удаляющую из вещественной матрицы A размера $M \times N$ строки с номерами от K_1 до K_2 включительно (предполагается, что $1 < K_1 \leq K_2$). Если $K_1 > M$, то матрица не изменяется; если $K_2 > M$, то удаляются строки матрицы с номерами от K_1 до M . Двумерный массив A и числа M, N являются входными и выходными параметрами. Используя

- процедуру `RemoveRows`, удалить из данной матрицы A размера $M \times N$ строки с номерами от K_1 до K_2 и вывести размер полученной матрицы и ее элементы.
- 24) Описать процедуру `RemoveCols(A, M, N, K1, K2)`, удаляющую из вещественной матрицы A размера $M \times N$ столбцы с номерами от K_1 до K_2 включительно (предполагается, что $1 < K_1 \leq K_2$). Если $K_1 > N$, то матрица не изменяется; если $K_2 > N$, то удаляются столбцы матрицы с номерами от K_1 до N . Двумерный массив A и числа M, N являются входными и выходными параметрами. Используя процедуру `RemoveCols`, удалить из данной матрицы A размера $M \times N$ столбцы с номерами от K_1 до K_2 и вывести размер полученной матрицы и ее элементы.
- 25) Описать процедуру `RemoveRowCol(A, M, N, K, L)`, удаляющую из вещественной матрицы A размера $M \times N$ строку и столбец, которые содержат элемент $A_{K,L}$ (предполагается, что $M > 1$ и $N > 1$; если $K > M$ или $L > N$, то матрица не изменяется). Двумерный массив A и числа M, N являются входными и выходными параметрами. Дана матрица A размера $M \times N$ и числа K, L . Применить к матрице A процедуру `RemoveRowCol` и вывести размер полученной матрицы и ее элементы.
- 26) Описать процедуру `SortCols(A, M, N)`, выполняющую сортировку по возрастанию столбцов целочисленной матрицы A размера $M \times N$ (столбцы сравниваются *лексикографически*: если первые элементы столбцов различны, то меньшим считается столбец, содержащий меньший первый элемент; если первые элементы столбцов равны, то анализируются их вторые элементы и т. д.). Двумерный массив A является входным и выходным параметром. Используя процедуру `SortCols`, отсортировать столбцы данной матрицы A размера $M \times N$.

Строки

- 27) Описать функцию `FillStr(S, N)` строкового типа, возвращающую строку длины N , заполненную повторяющимися копиями строки-шаблона S (последняя копия строки-шаблона может входить в результирующую строку частично). Используя эту функцию, сформировать по данному числу N и пяти данным строкам-шаблонам пять результирующих строк длины N .
- 28) Описать процедуру `UpCaseRus(S)`, преобразующую все строчные русские буквы строки S в прописные (остальные символы строки S не изменяются). Строка S является входным и выходным параметром. Используя процедуру `UpCaseRus`, преобразовать пять данных строк.
- 29) Описать процедуру `LowCaseRus(S)`, преобразующую все прописные русские буквы строки S в строчные (остальные символы строки S не изменяются). Строка S является входным и выходным параметром. Используя процедуру `LowCaseRus`, преобразовать пять данных строк.
- 30) Описать процедуру `TrimLeftC(S, C)`, удаляющую в строке S начальные символы, совпадающие с символом C . Строка S является входным и выходным параметром. Дан символ C и пять строк. Используя процедуру `TrimLeftC`, преобразовать данные строки.
- 31) Описать процедуру `TrimRightC(S, C)`, удаляющую в строке S конечные символы, совпадающие с символом C . Строка S является входным и выходным параметром. Дан символ C и пять строк. Используя процедуру `TrimRightC`, преобразовать данные строки.

- 32) Описать функцию $\text{InvertStr}(S, K, N)$ строкового типа, возвращающую *инвертированную подстроку* строки S , содержащую в обратном порядке N символов строки S , начиная с ее K -го символа. Если K превосходит длину строки S , то возвращается пустая строка; если длина строки меньше $K + N$, то инвертируются все символы строки, начиная с ее K -го символа. Вывести значения функции InvertStr для данной строки S и каждой из трех пар положительных целых чисел: (K_1, N_1) , (K_2, N_2) , (K_3, N_3) .
- 33) Описать функцию $\text{PosSub}(S_0, S, K, N)$ целого типа, возвращающую номер позиции, начиная с которой в строке S содержится первое вхождение строки S_0 , причем анализируются только N символов строки S , начиная с ее K -го символа (таким образом, PosSub обеспечивает *поиск в подстроке*). Если K превосходит длину строки S , то возвращается 0, если длина строки меньше $K + N$, то анализируются все символы строки, начиная с ее K -го символа. Если в требуемой подстроке строки S вхождения S_0 отсутствуют, то функция возвращает 0. Вывести значения функции PosSub для данных строк S_0, S и каждой из трех пар положительных целых чисел: (K_1, N_1) , (K_2, N_2) , (K_3, N_3) .
- 34) Описать функцию $\text{PosLast}(S_0, S)$ целого типа, возвращающую номер позиции, начиная с которой в строке S содержится последнее вхождение подстроки S_0 . Считать, что перекрывающихся вхождений подстрок S_0 строка S не содержит. Если в строке S отсутствуют подстроки S_0 , то функция возвращает 0. Вывести значения этой функции для пяти данных пар строк S_0 и S .
- 35) Описать функцию $\text{PosK}(S_0, S, K)$ целого типа, возвращающую номер позиции, начиная с которой в строке S содержится K -е вхождение подстроки S_0 ($K > 0$). Если количество вхождений S_0 в строке S меньше K , то функция возвращает 0. Считать, что перекрывающихся вхождений подстрок S_0 строка S не содержит. Вывести значения этой функции для пяти данных троек: S_0, S и K .
- 36) Описать функцию $\text{WordK}(S, K)$ строкового типа, возвращающую K -е слово строки S (*словом* считается набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки). Если количество слов в строке меньше K , то функция возвращает пустую строку. Используя эту функцию, выделить из данной строки S слова с данными номерами K_1, K_2, K_3 .
- 37) Описать процедуру $\text{SplitStr}(S, W, N)$, которая формирует по данной строке S массив W слов, входящих в S (массив W и его размер N являются выходными параметрами). *Словом* считается набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки; предполагается, что строка S содержит не более 10 слов. Используя функцию SplitStr , найти количество слов N , содержащихся в данной строке S , и сами эти слова.
- 38) Описать функцию $\text{DecompressStr}(S)$ строкового типа, восстанавливающую строку, сжатую процедурой CompressStr . Параметр S содержит сжатую строку; восстановленная строка является возвращаемым значением функции. С помощью функции DecompressStr восстановить пять данных сжатых строк.
- 39) Описать функцию $\text{DecToBin}(N)$ строкового типа, возвращающую строковое представление целого неотрицательного числа N в двоичной системе счисления. Результирующая строка состоит из символов «0»–«1» и не содержит ведущих нулей (за исключением представления числа 0). Используя эту функцию, получить двоичные представления пяти данных чисел.

- 40) Описать функцию $\text{DecToHex}(N)$ строкового типа, возвращающую строковое представление целого неотрицательного числа N в 16-ричной системе счисления. Результирующая строка состоит из символов «0»–«9», «A»–«F» и не содержит ведущих нулей (за исключением представления числа 0). Используя эту функцию, получить 16-ричные представления пяти данных чисел.
- 41) Описать функцию $\text{BinToDec}(S)$ целого типа, определяющую целое неотрицательное число по его строковому представлению S в двоичной системе счисления. Параметр S имеет строковый тип, состоит из символов «0»–«1» и не содержит ведущих нулей (за исключением значения «0»). Используя эту функцию, вывести пять чисел, для которых даны их двоичные представления.
- 42) Описать функцию $\text{HexToDec}(S)$ целого типа, определяющую целое неотрицательное число по его строковому представлению S в 16-ричной системе счисления. Параметр S имеет строковый тип, состоит из символов «0»–«9», «A»–«F» и не содержит ведущих нулей (за исключением значения «0»). Используя эту функцию, вывести пять чисел, для которых даны их 16-ричные представления.
- 43) Описать функцию $\text{IntFileSize}(S)$ целого типа, возвращающую количество элементов в файле целых чисел с именем S . Если файл не существует, то функция возвращает -1 . С помощью этой функции найти количество элементов в трех файлах с данными именами.

Файлы

- 44) Описать функцию $\text{LineCount}(S)$ целого типа, возвращающую количество строк в текстовом файле с именем S . Если файл не существует, то функция возвращает -1 . С помощью этой функции найти количество строк в трех файлах с данными именами.
- 45) Описать процедуру $\text{InvertIntFile}(S)$, меняющую порядок следования элементов файла целого типа с именем S на противоположный. Если файл не существует или содержит менее двух элементов, то процедура не выполняет никаких действий. Обработать с помощью этой процедуры три файла с данными именами.
- 46) Описать процедуру $\text{RemoveLineNumbers}(S)$, удаляющую из начала каждой строки существующего текстового файла с именем S ее порядковый номер, добавленный процедурой AddLineNumbers , а также пробелы, отделяющие номер от последующего текста. Если строки не содержат номеров, то процедура не выполняет никаких действий. Применить эту процедуру к файлу с данным именем.
- 47) Описать процедуру $\text{SplitIntFile}(S_0, K, S_1, S_2)$, копирующую первые K (> 0) элементов существующего файла целых чисел с именем S_0 в новый файл целых чисел с именем S_1 , а остальные элементы — в новый файл целых чисел с именем S_2 . Один из созданных файлов может остаться пустым. Применить эту процедуру к файлу с данным именем S_0 , используя указанные значения K, S_1 и S_2 .
- 48) Описать процедуру $\text{SplitText}(S_0, K, S_1, S_2)$, копирующую первые K ($K \neq 0$) строк существующего текстового файла с именем S_0 в новый текстовый файл с именем S_1 , а остальные строки — в новый текстовый файл с именем S_2 . Один из созданных файлов может остаться пустым. Применить эту процедуру к файлу с данным именем S_0 , используя указанные значения K, S_1 и S_2 .

- 49) Описать процедуру $\text{StringFileToText}(S)$, преобразующую двоичный строковый файл с именем S в текстовый файл с тем же именем. Используя эту процедуру, преобразовать два данных строковых файла с именами S_1 и S_2 в текстовые.
- 50) Описать процедуру $\text{TextToStringFile}(S)$, преобразующую текстовый файл с именем S в двоичный строковый файл с тем же именем. Используя эту процедуру, преобразовать два данных текстовых файла с именами S_1 и S_2 в строковые.
- 51) Описать процедуру $\text{EncodeText}(S, K)$, которая шифрует текстовый файл с именем S , выполняя циклическую замену каждой русской буквы на букву того же регистра, расположенную в алфавите на K -й позиции после шифруемой буквы ($0 < K < 10$). Например, при $K = 3$ «А» перейдет в «Г», «я» — в «в». Букву «ё» в алфавите не учитывать, считая, что за буквой «е» сразу идет «ж». Символы, не являющиеся русскими буквами, при шифровании не изменять. Используя эту процедуру и зная кодовое смещение K , зашифровать файл с указанным именем.
- 52) Описать процедуру $\text{DecodeText}(S, K)$, которая дешифрует текстовый файл с именем S , зашифрованный с использованием кодового смещения K . Используя эту процедуру и зная кодовое смещение K , расшифровать файл с указанным именем.

Записи

При вводе и выводе каждой даты вначале указывается день, затем номер месяца, затем год. При вводе каждой точки в заданиях Param64–Param70 вначале указывается ее абсцисса (x -координата), затем ее ордината (y -координата).

- 53) Используя тип TDate и функцию LeapYear , описать функцию $\text{DaysInMonth}(D)$ целого типа с параметром типа TDate , которая возвращает количество дней для месяца, указанного в дате D . Вывести значение функции DaysInMonth для пяти данных дат (предполагается, что все даты являются правильными).
- 54) Используя тип TDate и функцию DaysInMonth , описать функцию $\text{CheckDate}(D)$ целого типа с параметром типа TDate , которая проверяет правильность даты, указанной в параметре D . Если дата D является правильной, то функция возвращает 0; если в дате указан неверный номер месяца, то функция возвращает 1; если в дате указан неверный день для данного месяца, то возвращается 2. Вывести значение функции CheckDate для пяти данных дат.
- 55) Используя тип TDate и функции DaysInMonth и CheckDate , описать процедуру $\text{PrevDate}(D)$ с параметром типа TDate , которая преобразует дату D к предыдущей дате (если дата D является неправильной, то она не изменяется). Запись D является входным и выходным параметром. Применить процедуру PrevDate к пяти данным датам.
- 56) Используя тип TDate и функции DaysInMonth и CheckDate , описать процедуру $\text{NextDate}(D)$ с параметром типа TDate , которая преобразует дату D к следующей дате (если дата D является неправильной, то она не изменяется). Запись D является входным и выходным параметром. Применить процедуру NextDate к пяти данным датам.
- 57) Описать тип TPoint — запись с полями вещественного типа X и Y (координаты точки на плоскости) — и функцию $\text{Leng}(A, B)$ вещественного типа, находящую длину отрезка AB на плоскости по координатам его концов:

$$|AB| = \sqrt{(A.X - B.X)^2 + (A.Y - B.Y)^2}$$

(A и B — параметры типа TPoint). С помощью этой функции найти длины отрезков AB , AC , AD , если даны координаты точек A , B , C , D .

58) Используя тип TPoint и функцию Leng, описать тип TTriangle — запись с полями A , B , C типа TPoint (вершины треугольника) — и функцию Perim(T) вещественного типа, находящую периметр треугольника T (T — параметр типа TTriangle). С помощью этой функции найти периметры треугольников ABC , ABD , ACD , если даны координаты точек A , B , C , D .

59) Используя типы TPoint, TTriangle и функции Leng и Perim, описать функцию Area(T) вещественного типа, находящую площадь треугольника T (T — параметр типа TTriangle) по формуле Герона:

$$S_{ABC} = \sqrt{p \cdot (p - |AB|) \cdot (p - |AC|) \cdot (p - |BC|)},$$

где p — полупериметр. С помощью этой функции найти площади треугольников ABC , ABD , ACD , если даны координаты точек A , B , C , D .

60) Используя типы TPoint, TTriangle и функции Leng и Area, описать функцию Dist(P , A , B) вещественного типа (P , A , B — параметры типа TPoint), находящую расстояние $D(P, AB)$ от точки P до прямой AB по формуле

$$D(P, AB) = 2 \cdot S_{PAB} / |AB|,$$

где S_{PAB} — площадь треугольника PAB . С помощью этой функции найти расстояния от точки P до прямых AB , AC , BC , если даны координаты точек P , A , B , C .

61) Используя типы TPoint, TTriangle и функцию Dist, описать процедуру Heights(T , h_1 , h_2 , h_3), находящую высоты h_1 , h_2 , h_3 треугольника T (T — входной параметр типа TTriangle, h_1 , h_2 , h_3 — выходные вещественные параметры), проведенные соответственно из вершин $T.A$, $T.B$, $T.C$. С помощью этой процедуры найти высоты треугольников ABC , ABD , ACD , если даны координаты точек A , B , C , D .

62) Используя тип TPoint и функцию Leng, описать функцию PerimN(P , N) вещественного типа, находящую периметр N -угольника, вершины которого (в порядке их обхода) передаются в массиве P размера N (> 2) с элементами типа TPoint. С помощью этой функции найти периметры трех многоугольников, если дано число их сторон и координаты их вершин.

63) Используя типы TPoint, TTriangle и функцию Area, описать функцию AreaN(P , N) вещественного типа, находящую площадь выпуклого N -угольника, вершины которого (в порядке их обхода) передаются в массиве P размера N (> 2) с элементами типа TPoint. С помощью этой функции найти площади трех многоугольников, если дано число их сторон и координаты их вершин.

17. Рекурсия: группа Recur

1. Простейшие рекурсивные алгоритмы

Задания этого пункта можно легко решить и *без использования рекурсии*. Данное обстоятельство связано с тем, что в заданиях рассматриваются *простейшие* примеры рекурсии, легко сводимые к итерационным алгоритмам. Более того, в некоторых случаях непосредственные вычисления по рекурсивным формулам оказываются весьма неэффективными. Однако именно на подобных примерах проще всего получить первоначальные навыки разработки рекурсивных алгоритмов.

1) Описать рекурсивную функцию $\text{Fact}(N)$ вещественного типа, вычисляющую значение *факториала*

$$N! = 1 \cdot 2 \cdot \dots \cdot N$$

($N > 0$ — параметр целого типа). С помощью этой функции вычислить факториалы пяти данных чисел.

PascalABC.NET	C++
<pre>Var N, i: integer; function Fact(N: integer): real; begin if N > 1 then Fact:= N * Fact(N - 1) else Fact := 1; end; begin for i := 1 to 5 do begin Write('N: '); Readln(N); Writeln(Fact(N)); end; end.</pre>	<pre>#include <iostream> using namespace std; long fact(int n) { if (n <= 1) return 1; return n*fact(n - 1); } int main() { int i, n; for (i = 1; i <= 5; i++) { cin >> n; long f = fact(n); cout << f << endl; } system("pause"); return 0; }</pre>
Python	C#
<pre>def fact(n): if n<=1: return 1; else: return n*fact(n-1) for i in range(1,5): n=int(input("->")) print(fact(n))</pre>	<pre>static long fact(int n){ if (n<=1) return 1; else return n*fact(n-1); } static void Main(string[] args){ int i, n; for (i=0; i<5; i++) { n=Convert.ToInt32(Console.ReadLine()); long m=fact(n); Console.WriteLine(m); } Console.ReadKey(); }</pre>

Java	Go
<pre> static long fact(int n){ if (n<=1) return 1; else return n*fact(n-1); } public static void main(String[] args){ Scanner sc=new Scanner(System.in); for (int i=1; i<=5; i++){ int n=sc.nextInt(); long m=fact(n); System.out.println(m); } } </pre>	<pre> package main import "fmt" func fact(n int) int{ if (n<=1){ return 1; }else{ return n*fact(n-1); } } func main() { n:=0 for i:=1; i<=5; i++){ fmt.Scan(&n) m:=fact(n) fmt.Println(m) } } </pre>
Kotlin	
<pre> import java.util.Scanner fun fact(n: Long) : Long { return if (n <= 1) { n } else { n * fact(n - 1) } } fun main(args: Array<String>) { val reader = Scanner(System.`in`) for (i in (1..5)) { var n = reader.nextLong(); println(fact(n)) } } </pre>	
VB.NET	
<pre> Function Factorial(x As Integer) As Integer If (x = 1) Then Return 1 Else Return x * Factorial(x - 1) End If End Function Sub Main() Dim i, n, m As Integer For i = 0 To 5 n = Convert.ToInt32(Console.ReadLine()) m = Factorial(n) Console.WriteLine(m) Next Console.ReadKey() End Sub </pre>	

2) Описать рекурсивную функцию PowerN(X, N) вещественного типа, находящую значение N-й степени числа X по формулам:

$$X^0 = 1,$$

$$X^N = (X^{N/2})^2 \text{ при четных } N > 0,$$

$$X^N = X \cdot X^{N-1} \text{ при нечетных } N > 0,$$

$$X^N = 1/X^{-N} \text{ при } N < 0$$

($X \neq 0$ — вещественное число, N — целое; в формуле для четных N должна использоваться операция *целочисленного деления*). С помощью этой функции найти значения X^N для данного X при пяти данных значениях N .

PascalABC.NET
<pre> Var N, i: integer; X: Real; function PowerN(X: Real; N: integer): real; begin if N <> 0 then if N < 0 then PowerN := 1 / PowerN(X, -N) else if N mod 2 = 0 then PowerN := sqr(PowerN(X, N div 2)) else PowerN := X * PowerN(X, N - 1) else PowerN := 1; end; begin Write('X: '); Readln(X); for i := 1 to 5 do begin Write('N: '); Readln(N); Writeln(PowerN(X, N)); end; end. </pre>
Python
<pre> def PowerN(X, N): if N != 0: if N < 0: return 1 / PowerN(X, -N) else: if N / 2 == 0: return sqr(PowerN(X, N / 2)) else: return X * PowerN(X, N - 1) </pre>

```

else:
    return 1
print('X: ')
X = float(input())
for i in range(0, 5):
    print('N: ')
    N = int(input())
    print(PowerN(X, N))

```

C++

```

#include <iostream>
using namespace std;
float PowerN(float x, int n) {
    if (n != 0)
        if (n < 0)
            return 1.0 / pow(x, -n);
        else
            if (n % 2 == 0)
                return PowerN(x, n / 2)*PowerN(x, n / 2);
            else
                return x * PowerN(x, n - 1);
    else
        return 1;
}
int main() {
    int i, n;
    float f, x;
    cin >> x;
    for (i = 1; i <= 5; i++) {
        cin >> n;
        long f = PowerN(x, n);
        cout << f << endl;
    }
    system("pause");
    return 0;
}

```

3) Описать рекурсивную функцию $Combin1(N, K)$ целого типа, находящую $C(N, K)$ — число сочетаний из N элементов по K — с помощью рекуррентного соотношения:

$$C(N, 0) = C(N, N) = 1, C(N, K) = C(N - 1, K) + C(N - 1, K - 1) \quad \text{при } 0 < K < N.$$

Параметры функции — целые числа; $N > 0, 0 \leq K \leq N$. Дано число N и пять различных значений K . Вывести числа $C(N, K)$ вместе с количеством рекурсивных вызовов функции $Combin1$, потребовавшихся для их нахождения.

```

PascalABC.NET

Var
    N, K, i, Num: integer;

function CombiN_1(N, K: integer): integer;
begin

```

```

if (K = 0) or (K = N) then
    CombiN_1 := 1
else
    CombiN_1:=CombiN_1(N-1,K)+CombiN_1(N - 1, K - 1);
    inc(Num);
end;

begin
    Write('N: ');
    Readln(N);
    for i := 1 to 5 do
        begin
            Num := 0;
            Write('K: ');
            Readln(K);
            Writeln(CombiN_1(N, K));
            Writeln(Num);
        end;
    end.

```

C++

```

#include <iostream>
using namespace std;
int CombiN_1(int n, int k, int &Num) {
    if ((k = 0) || (k = n))
        return 1;
    else
        return CombiN_1(n-1,k,Num)+CombiN_1(n-1,k-1,Num);
    Num++;
}

int main() {
    int n, k, i, Num
    cout << "n="; cin >> n;
    for (i = 1; i <= 5; i++) {
        cout << "k="; cin >> k;
        cout << CombiN_1(n, k, Num) << endl;;
        cout << Num << endl;
    }
    system("pause");
    return 0;
}

```

- 4) Описать рекурсивную функцию $\text{RootK}(X, K, N)$ вещественного типа, находящую приближенное значение корня K -й степени из числа X по формуле:

$$Y_0 = 1, \quad Y_{N+1} = Y_N - (Y_N - X/(Y_N)^{K-1})/K,$$

где Y_N обозначает $\text{RootK}(X, K, N)$ при фиксированных X и K . Параметры функции: $X (> 0)$ — вещественное число, $K (> 1)$ и $N (> 0)$ — целые. С помощью функции RootK найти для данного числа X приближенные значения его корня K -й степени при шести данных значениях N .

PascalABC.NET

```

Var
  X, K, N, i: integer;

function RootK(X: real; K, N: integer): real;
begin
  if N = 0 then RootK := 1
  else
    RootK := RootK(X, K, N - 1) - (RootK(X, K, N - 1) - X /
      power(RootK(X, K, N - 1), K - 1)) / K;
  end;

begin
  (X, K) := readInteger2('X,K:');
  for i := 1 to 6 do
    begin
      Write('N: ');
      Readln(N);
      Writeln(RootK(X, K, N));
    end;
end.

```

C++

```

#include <iostream>
#include <cmath>
using namespace std;
float RootK(float x, int k, int n)
{
  if (n == 0)
    return 1;
  else
    return RootK(x, k, n - 1) - (RootK(x, k, n - 1) - x
      / pow(RootK(x, k, n - 1), k - 1)) / k;
}
int main()
{
  int i, k, n;
  float x;
  cin >> x >> k;
  for (i = 1; i <= 5; i++) {
    cout << "n="; cin >> n;
    cout << RootK(x,k,n) << endl;;
  }
  system("pause");
  return 0;
}

```

- 5) Описать рекурсивную функцию DigitSum(K) целого типа, которая находит сумму цифр целого числа K , не используя оператор цикла. С помощью этой функции найти суммы цифр для пяти данных целых чисел.

PascalABC.NET	C++
<pre> Var i, K: integer; function DigitSum(A: integer): integer; begin if (A Div 10) = 0 then DigitSum := (A mod 10) else DigitSum := (A mod 10) + DigitSum(A Div 10); end; begin for i := 1 to 5 do begin Write('K: '); Readln(K); Writeln(DigitSum(K)); end; end. </pre>	<pre> #include <iostream> using namespace std; int DigitSum(int a) { if ((a / 10) == 0) return a % 10; else return (a % 10)+DigitSum(a/10); } int main() { int i, n; for (i = 1; i <= 5; i++) { cout << "n="; cin >> n; cout << DigitSum(n)<< endl; } system("pause"); return 0; } </pre>

- б) Описать рекурсивную функцию $\text{MaxInt}(A, N)$ целого типа, которая находит максимальный элемент целочисленного массива A размера N ($1 \leq N \leq 10$), не используя оператор цикла. С помощью этой функции найти максимальные элементы массивов A, B, C размера N_A, N_B, N_C соответственно.

PascalABC.NET
<pre> Type ar = array[1..10]of integer; var NA, NB, NC: integer; A, B, C: ar; function MaxInt(A: ar; N: integer): integer; begin if N < 2 then MaxInt := A[N] else if A[N] > MaxInt(A, N - 1) then MaxInt := A[N] else MaxInt := MaxInt(A, N - 1); end; procedure InputArray(var A: ar; N: integer); var i: integer; begin for i := 1 to N do begin Write(i, ': '); Readln(A[i]); end; end; </pre>

```

    end;
end;

begin
    Write('NA: ');
    Readln(NA);
    InputArray(A, NA);
    Writeln(MaxInt(A, NA));

    Write('NB: ');
    Readln(NB);
    InputArray(B, NB);
    Writeln(MaxInt(B, NB));

    Write('NC: ');
    Readln(NC);
    InputArray(C, NC);
    Writeln(MaxInt(C, NC));
end.

```

C++

```

#include <iostream>
using namespace std;
int MaxInt(int a[], int n) {
    if (n < 2)
        return a[n];
    else
        if (a[n] > MaxInt(a, n - 1))
            return a[n];
        else
            return MaxInt(a, n - 1);
}
void initAr(int x[], int n) {
    for (int i = 0; i < n; i++)
        x[i] = rand() % 50;
}
void ShowArr(int x[], int n) {
    for (int i = 0; i < n; i++)
        cout << x[i] << " ";
    cout << endl;
}
int main() {
    int i, Na, Nb, Nc, *a,*b, *c;
    cout << "Na="; cin >> Na;
    a = new int[Na + 1];
    initAr(a, Na);
    ShowArr(a, Na);
    cout << MaxInt(a, Na)<<endl;
    delete a;
    cout << "Nb="; cin >> Nb;
    b = new int[Nb + 1];

```



```

    initAr(b, Nb);
    ShowArr(b, Nb);
    cout << MaxInt(b, Nb)<<endl;
    delete b;
    cout << "Nc="; cin >> Nc;
    c = new int[Nc + 1];
    initAr(c, Nc);
    ShowArr(c, Nc);
    cout << MaxInt(c, Nc) << endl;
    delete c;
    system("pause");
    return 0;
}

```

2. Разбор выражений

Во всех заданиях данного пункта предполагается, что исходные строки, определяющие выражения, не содержат пробелов. При выполнении заданий рекомендуется обойтись без операторов цикла.

7) Вывести значение целочисленного выражения, заданного в виде строки S.

Выражение определяется следующим образом:

<выражение>	::=	<цифра> <выражение> + <цифра>
		<выражение> - <цифра>

Решения

```

PascalABC.NET

Var
  S: String;

function Calculation(S: String): integer;
var
  Stemp: string;
begin
  Stemp := S;
  delete(Stemp, 1, 2);
  if S.Length <= 0 then
    Result := 0
  else
    if s[2] = '-' then
      Result := (ord(s[1]) - 48) - Calculation(Stemp)
    else
      Result := (ord(s[1]) - 48) + Calculation(Stemp);
end;

begin
  Write('S: ');
  Readln(S);
  S := S + ' ';
  Writeln(Calculation(S));
end.

```

```
C++
```

```

#include <iostream>
#include <string>
using namespace std;
int Calculation(string S) {
    string Stemp = S;
    if (S.length() > 1) {
        Stemp = Stemp.substr(2, S.length());
        if (Stemp.length() <= 0)
            return (int)S[0] - 48;
        else
            if (S[1] == '+')
                return ((int)S[0] - 48) + Calculation(Stemp);
            else
                return ((int)S[1] - 48) - Calculation(Stemp);
    }
    return (int)S[0] - 48;
}
int main() {
    string S;
    cin >> S;
    cout << Calculation(S);
    return 0;
}

```

8) Вывести значение целочисленного выражения, заданного в виде строки *S*.
 Выражение определяется следующим образом:

<выражение>	::=	<терм> <выражение> + <терм>
		<выражение> - <терм>
<терм>	::=	<элемент> <терм> * <элемент>
<элемент>	::=	<цифра> (<выражение>)

Решения

```
PascalABC.NET
```

```

function LenExpression(S: String): integer;
var
    len: integer;
begin
    if length(S) <= 0 then Result := 0
    else
        case s[length(S)] of
            ')':
                begin
                    delete(S, length(S), 1);
                    len := LenExpression(S);
                    delete(S, length(S) - len + 1, len);
                    Result := len + LenExpression(S) + 1;
                end;
            '(': Result := 1
        else

```

```

    begin
        delete(S, length(S), 1);
        Result := 1 + LenExpression(S);
    end;
end;

function Expression(S: String): integer;
var
    Res, len: integer;
begin
    Res := 0;
    if length(S) > 0 then
        begin
            case s[length(S)] of
                ')':
                    begin
                        delete(S, length(S), 1);
                        len := LenExpression(S);
                        Res := Expression(S);
                        delete(S, length(S) - len + 1, len);
                    end;
                '0'..'9':
                    begin
                        Res := (ord(S[length(S)]) - 48);
                        delete(S, length(S), 1);
                    end;
            end;
        if length(S) > 0 then
            case s[length(S)] of
                '-':
                    begin
                        delete(S, length(S), 1);
                        Result := Expression(S) - Res;
                    end;
                '+':
                    begin
                        delete(S, length(S), 1);
                        Result := Expression(S) + Res;
                    end;
                '*':
                    begin
                        delete(S, length(S), 1);
                        Result := Expression(S) * Res;
                    end;
                '(': Result := Res;
            end
        else Result := Res;
        end
    else Writeln('!!error!!');
end;

```

```

end;

var
  S: String;

begin
  Write('S: ');
  Readln(S);
  S := '(' + S + ')';
  Writeln('Exp: ', Expression(S));
end.

```

- 9) Вывести значение целочисленного выражения, заданного в виде строки S. Выражение определяется следующим образом:

<выражение>	::=	<цифра>
		(<выражение><знак><выражение>)
<знак>	::=	+ - *

Решения

```

PascalABC.NET

function LenExpression(S: String): integer;
var
  len: integer;
begin
  if length(S) <= 0 then Result := 0
  else
    case s[length(S)] of
      ')':
        begin
          delete(S, length(S), 1);
          len := LenExpression(S);
          delete(S, length(S) - len + 1, len);
          Result := len + LenExpression(S) + 1;
        end;
      '(': Result := 1
    else
      begin
        delete(S, length(S), 1);
        Result := 1 + LenExpression(S);
      end;
    end;
end;

function Expression(S: String): integer;
var
  Res, len: integer;
begin
  Res := 0;
  if length(S) > 0 then
    begin

```

```

case s[length(S)] of
  ')':
    begin
      delete(S, length(S), 1);
      len := LenExpression(S);
      Res := Expression(S);
      delete(S, length(S) - len + 1, len);
    end;
  '0'..'9':
    begin
      Res := (ord(S[length(S)]) - 48);
      delete(S, length(S), 1);
    end;
end;
if length(S) > 0 then
  case s[length(S)] of
    '-':
      begin
        delete(S, length(S), 1);
        Result := Expression(S) - Res;
      end;
    '+':
      begin
        delete(S, length(S), 1);
        Result := Expression(S) + Res;
      end;
    '*':
      begin
        delete(S, length(S), 1);
        Result := Expression(S) * Res;
      end;
    '(': Result := Res;
  end
  else Result := Res;
end
else Writeln('!!error!!');
end;
var
  S: String;
begin
  Write('S: ');
  Readln(S);
  S := '(' + S + ')';
  Writeln('Exp: ', Expression(S));
end.

```

3. Перебор с возвратом

10) Дано дерево глубины N , каждая внутренняя вершина которого имеет K (< 10) непосредственных потомков (нумеруются от 1 до K). Корень дерева имеет номер 0. Записать в текстовый файл с данным именем все возможные пути, ведущие от корня к листьям. Перебирать пути, начиная с «самого левого» и заканчивая «самым правым» (при этом первыми заменять конечные элементы пути).

Решения

```
PascalABC.NET

Var
  F: Text;
  K, N: Integer;
  FN: String;

procedure node(Ancestors: String; lvl: integer);
var
  F: Text;
  i: integer;
begin
  inc(lvl);
  if lvl = N + 1 then
  begin
    Assign(F, FN);
    Append(F);
    Writeln(F, Ancestors);
    Close(F);
  end
  else
  for i := 1 to K do
    node(Ancestors + chr(i + 48), lvl);
  end;
begin
  Write('N:');
  readln(N);
  Write('K:');
  readln(K);
  Write('FileName:');
  readln(FN);
  Assign(F, FN);
  Rewrite(F);
  Close(F);
  node('0', 0);
end.
```

11) Дано дерево глубины N , каждая внутренняя вершина которого имеет 3 непосредственных потомка: A с весом 1, B с весом 0 и C с весом -1 . Корень дерева D имеет вес 0. Записать в текстовый файл с данным именем все пути от корня к листьям, удовлетворяющие следующим условиям: суммарный вес элементов для любого начального отрезка пути неположителен, а суммарный вес

всех элементов пути равен 0. Порядок перебора путей такой же, как в задании Recur25.

Решения

```
PascalABC.NET

Var
  F: Text;
  N: Integer;
  FN: String;

procedure node(Ancestors: String; lvl, weight: integer);
var
  F: Text;
begin
  inc(lvl);
  if (lvl = N + 1) and (weight = 0) then
  begin
    Assign(F, FN);
    Append(F);
    Writeln(F, Ancestors);
    Close(F);
  end;
  if lvl <= N then
  begin
    if weight < 0 then
      node(Ancestors + 'A', lvl, weight + 1);
      node(Ancestors + 'B', lvl, weight);
      node(Ancestors + 'C', lvl, weight - 1);
    end;
  end;
begin
  Write('N:');
  readln(N);
  Write('FileName:');
  readln(FN);
  Assign(F, FN);
  Rewrite(F);
  Close(F);
  node('D', 0, 0);
end.
```

Задачи для самостоятельной работы

Простейшие рекурсивные алгоритмы

- 1) Описать рекурсивную функцию $\text{Fact2}(N)$ вещественного типа, вычисляющую значение *двойного факториала*

$$N!! = N \cdot (N-2) \cdot (N-4) \cdot \dots$$

($N > 0$ — параметр целого типа; последний сомножитель в произведении равен 2, если N — четное число, и 1, если N — нечетное). С помощью этой функции вычислить двойные факториалы пяти данных чисел.

- 2) Описать рекурсивную функцию $\text{Fib1}(N)$ целого типа, вычисляющую N -й элемент последовательности *чисел Фибоначчи* (N — целое число):

$$F_1 = F_2 = 1, \quad F_K = F_{K-2} + F_{K-1}, \quad K = 3, 4, \dots$$

С помощью этой функции найти пять чисел Фибоначчи с данными номерами, и вывести эти числа вместе с количеством рекурсивных вызовов функции Fib1 , потребовавшихся для их нахождения.

- 3) Описать рекурсивную функцию $\text{Fib2}(N)$ целого типа, вычисляющую N -й элемент последовательности *чисел Фибоначчи* (N — целое число):

$$F_1 = F_2 = 1, \quad F_K = F_{K-2} + F_{K-1}, \quad K = 3, 4, \dots$$

Считать, что номер N не превосходит 20. Для уменьшения количества рекурсивных вызовов по сравнению с функцией Fib1 создать вспомогательный массив для хранения *уже вычисленных* чисел Фибоначчи и обращаться к нему при выполнении функции Fib2 . С помощью функции Fib2 найти пять чисел Фибоначчи с данными номерами.

- 4) Описать рекурсивную функцию $\text{Combin2}(N, K)$ целого типа, находящую $C(N, K)$ — *число сочетаний* из N элементов по K — с помощью рекуррентного соотношения:

$$C(N, 0) = C(N, N) = 1, \quad C(N, K) = C(N-1, K) + C(N-1, K-1) \quad \text{при } 0 < K < N.$$

Параметры функции — целые числа; $N > 0, 0 \leq K \leq N$. Считать, что параметр N не превосходит 20. Для уменьшения количества рекурсивных вызовов по сравнению с функцией Combin1 описать вспомогательный двумерный массив для хранения *уже вычисленных* чисел $C(N, K)$ и обращаться к нему при выполнении функции Combin2 . С помощью функции Combin2 найти числа $C(N, K)$ для данного значения N и пяти различных значений K .

- 5) Описать рекурсивную функцию $\text{NOD}(A, B)$ целого типа, находящую наибольший общий делитель (НОД) двух натуральных чисел A и B , используя алгоритм Евклида:

$$\text{НОД}(A, B) = \text{НОД}(B, A \bmod B), \quad \text{если } B \neq 0; \quad \text{НОД}(A, 0) = A.$$

С помощью этой функции найти

$$\text{НОД}(A, B), \text{НОД}(A, C), \text{НОД}(A, D),$$

если даны числа A, B, C, D .

- 6) Описать рекурсивную функцию $\text{DigitCount}(S)$ целого типа, которая находит количество цифр в строке S , не используя оператор цикла. С помощью этой функции найти количество цифр в каждой из пяти данных строк.

- 7) Описать рекурсивную функцию $\text{Palindrom}(S)$ логического типа, возвращающую True , если строка S является *палиндромом* (то есть читается одинаково слева направо и справа налево), и False в противном случае. Оператор цикла в теле функции не использовать. Вывести значения функции Palindrom для пяти данных строк.

Перебор с возвратом

- 8) Дано дерево глубины N , каждая внутренняя вершина которого имеет K ($K < 10$) непосредственных потомков (нумеруются от 1 до K). Корень дерева имеет номер 0. Записать в текстовый файл с данным именем все пути, ведущие от корня к листьям и удовлетворяющие следующему условию: никакие соседние элементы пути не нумеруются одной и той же цифрой. Порядок перебора путей такой же.
- 9) Дано дерево глубины N (N — четное), каждая внутренняя вершина которого имеет 2 непосредственных потомка: A с весом 1 и B с весом -1 . Корень дерева C имеет вес 0. Записать в текстовый файл с данным именем все пути от корня к листьям, удовлетворяющие следующему условию: суммарный вес элементов пути равен 0. Порядок перебора путей такой же.
- 10) Дано дерево глубины N того же типа, что и в задании Recur27. Записать в текстовый файл с данным именем все пути от корня к листьям, удовлетворяющие следующему условию: суммарный вес элементов для любого начального отрезка пути неотрицателен. Порядок перебора путей такой же.
- 11) Дано дерево глубины N того же типа, что и в задании Recur29. Записать в текстовый файл с данным именем все пути от корня к листьям, удовлетворяющие следующим условиям: никакие соседние элементы пути не обозначаются одной и той же буквой, а суммарный вес всех элементов пути равен 0. Порядок перебора путей такой же.

Разбор выражений

Во всех заданиях данного пункта предполагается, что исходные строки, определяющие выражения, не содержат пробелов. При выполнении заданий рекомендуется обойтись без операторов цикла.

- 12) Вывести значение целочисленного выражения, заданного в виде строки S . Выражение определяется следующим образом:

<выражение>	::=	<терм> <выражение> + <терм>
		<выражение> - <терм>
<терм>	::=	<цифра> <терм> * <цифра>

Вывести значение целочисленного выражения, заданного в виде строки S . Выражение определяется следующим образом (функция M возвращает максимальный из своих параметров, а функция m — минимальный):

<выражение>	::=	<цифра> $M(<выражение>, <выражение>)$
		$m(<выражение>, <выражение>)$

- 13) Проверить правильность выражения, заданного в виде непустой строки S . Если выражение составлено правильно, то вывести True, иначе вывести False.
- 14) Проверить правильность выражения, заданного в виде непустой строки S . Если выражение составлено правильно, то вывести 0, в противном случае вывести номер первого ошибочного (или лишнего) символа в строке S .
- 15) Вывести значение логического выражения, заданного в виде строки S . Выражение определяется следующим образом («Т» — True, «F» — False):

<выражение>	::=	T F And(<выражение>, <выражение>)
		Or(<выражение>, <выражение>)

- 16) Вывести значение целочисленного выражения, заданного в виде строки S . Выражение определяется следующим образом (функция M возвращает максимальный из своих параметров, а функция m — минимальный):

<выражение>	::=	<цифра> $M(\langle \text{параметры} \rangle)$ $m(\langle \text{параметры} \rangle)$
<параметры>	::=	<выражение> <выражение> , <параметры>

- 17) Вывести значение логического выражения, заданного в виде строки S . Выражение определяется следующим образом («Т» — True, «F» — False):

<выражение>	::=	T F And(<параметры>) Or(<параметры>)
<параметры>	::=	<выражение> <выражение> , <параметры>

- 18) Вывести значение логического выражения, заданного в виде строки S . Выражение определяется следующим образом («Т» — True, «F» — False):

<выражение>	::=	T F And(<параметры>)
		Or(<параметры>) Not(<выражение>)
<параметры>	::=	<выражение> <выражение> , <параметры>

18. Указатели и динамические структуры данных: группа **Pointer**

Все числа, упоминаемые в заданиях данной группы, являются *целыми*. Все указатели имеют тип PNode и указывают на записи типа TNode.

На языке Pascal типы PNode и TNode описываются следующим образом:

```
type
  PNode = ^TNode;
  TNode = record
    Data: Integer;
    Next: PNode;
    Prev: PNode;
  end;
```

Приведем также описание этих типов на языке C++:

```
struct TNode
{
  int Data;
  TNode* Next;
  TNode* Prev;
};
typedef TNode* PNode;
```

В заданиях на стеки и очереди при работе с записями типа TNode используются только поля Data и Next в заданиях на двусвязные списки используются все поля записи TNode.

Так как переменные типа «указатель» предназначены для хранения *адресов*, в формулировках заданий слова «указатель» (на элемент данных) и «адрес» (элемента данных) используются как синонимы. Имя *nil* для *нулевого указателя* в формулировках заданий заимствовано из языка Pascal. В языке C++ в качестве нулевого указателя можно использовать обычное число 0 или макроопределение NULL.

1. Стеки

- 1) Дан адрес P_1 записи типа TNode, содержащей поле Data (целого типа) и поле Next (типа PNode — указателя на TNode). Эта запись связана полем Next со следующей записью того же типа. Вывести значения полей Data обеих записей, а также адрес P_2 следующей записи.

PascalABC.NET
<pre>Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; end; var P1: PNode; begin New(P1); Write('P1.data: ');</pre>

```

Readln(P1^.Data);
New(P1^.Next);
Write('P2.data: ');
Readln(P1^.Next^.Data);
Writeln('P1.data:', P1^.Data);
Writeln('P1.next:', P1^.Next);
Writeln('P2.data:', P1^.Next^.Data);
Writeln('P2.next:', P1^.Next^.Next);
end.

```

C++

```

struct TNode{
    int Data;
    TNode* Next;
};
int main(){
    TNode *P1=new TNode, *P2=new TNode;
    P1->Data = 7;
    P1->Next = P2;
    P2->Data = 8;
    P2->Next = NULL;
    cout << P1->Data << " " << P1->Next->Data << " " << &P2 << endl;
    system("pause");
    return 0;
}

```

2) Дан адрес P_1 записи типа TNode. Эта запись связана полем Next со следующей записью того же типа, она, в свою очередь, — со следующей, и так далее до записи, поле Next которой равно nil (таким образом, возникает *цепочка* связанных записей). Вывести значения полей Data для всех элементов цепочки, *длину цепочки* (то есть число ее элементов) и адрес ее последнего элемента.

PascalABC.NET

```

Type
PNode = ^TNode;
TNode = record
    Data: Integer;
    Next: PNode;
end;
procedure CreateStack(var P1: PNode);
var
    p, q: PNode;
    i, N: integer;
begin
    Write('N:');
    Readln(N);

    New(P1);
    Write('data: ');
    Readln(P1^.Data);
    P1^.Next := nil;

    q := P1;

```

```

for i := 1 to N - 1 do
begin
    new(p);
    Write('data: ');
    Readln(P^.Data);
    p^.Next := nil;
    q^.Next := p;
    q := q^.Next
end;
end;

procedure ViewStack(P1: PNode);
var
    p: PNode;
    len: integer;
begin
    len := 1;
    p := p1;
    while p <> nil do
    begin
        inc(len);
        Writeln(p^.Data);
        if p^.Next = nil then
            writeln('last address: ', &p^.next);
        p := p^.next;
    end;
    writeln('len: ', len);
end;

var
    P1: PNode;

begin
    CreateStack(P1);
    ViewStack(P1);
end.

```

C++

```

#include <iostream>
using namespace std;
struct Node {
    int data;
    struct Node *next;
};
struct Node* top = NULL;
void push(int val) {
    struct Node* newnode = new Node;
    newnode->data = val;
    newnode->next = top;
    top = newnode;
}

```

```

void pop() {
    if (top == NULL)
        cout << "Stack Underflow" << endl;
    else {
        cout << "The popped element is " << top->data << endl;
        top = top->next;
    }
}
void display() {
    struct Node* ptr;
    if (top == NULL)
        cout << "stack is empty";
    else {
        ptr = top;
        cout << "Stack elements are: ";
        while (ptr != NULL) {
            cout << ptr->data << " ";
            ptr = ptr->next;
        }
        cout << &ptr<<endl;
    }
}
int main(){
    int val, n, i;
    cout<<"n="; cin >> n;
    for (i = 0; i < n; i++) {
        cin >> val;
        push(val);
    }
    display();
    system("pause");
    return 0;
}

```

3) Дано число D и указатель P_1 на вершину непустого стека. Добавить элемент со значением D в стек и вывести адрес P_2 новой вершины стека.

PascalABC.NET
Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; end;
var P1, PAdd: PNode; D: integer;
begin Write('D: '); Readln(D);

```

New(P1);
New(PAdd);
PAdd^.Data := D;
P1^.Next := PAdd;
Writeln('P1.next:', P1^.Next);
end.

```

- 4) Дан указатель P_1 на вершину стека, содержащего не менее десяти элементов. Извлечь из стека первые девять элементов и вывести их значения. Вывести также адрес новой вершины стека. После извлечения элементов из стека освободить память, которую они занимали.

```

PascalABC.NET
Type
PNode = ^TNode;
TNode = record
    Data: Integer;
    Next: PNode;
end;

procedure CreateStack(var P1: PNode);
var
    p: PNode;
    i, N: integer;
begin
    Write('N=10:');
    Readln(N);
    p1 := nil;
    for i := 1 to N do
    begin
        new(p);
        Write('num: ');
        Readln(P^.Data);
        p^.Next := P1;
        P1 := p;
    end;
end;

function pop(var P1: PNode): integer;
var
    head: PNode;
begin
    Result := p1^.Data;
    head := P1^.Next;
    Dispose(P1);
    P1 := head;
end;

var
    P1: PNode;

```

```

i: integer;

begin
  CreateStack(P1);
  for i := 1 to 10 do
    Writeln(pop(P1));
  end.

```

- 5) Даны указатели P_1 и P_2 на вершины двух непустых стеков. Переместить все элементы из первого стека во второй (в результате элементы первого стека будут располагаться во втором стеке в порядке, обратном исходному) и вывести адрес новой вершины второго стека. Операции выделения и освобождения памяти не использовать.

```

PascalABC.NET

Type
  PNode = ^TNode;
  TNode = record
    Data: Integer;
    Next: PNode;
  end;

procedure CreateStack(var P1: PNode);
var
  p: PNode;
  i, N: integer;
begin
  Write('N:');
  Readln(N);
  p1 := nil;
  for i := 1 to N do
    begin
      new(p);
      Write('num: ');
      Readln(P^.Data);
      p^.Next := P1;
      P1 := p;
    end;
  end;

procedure Cat(var P1: PNode; P2: PNode);
var
  Pt1: PNode;
begin
  Pt1 := P1;
  while (Pt1^.Next <> nil) do
    Pt1 := Pt1^.Next;
    pt1^.Next := P2;
  end;
end;

```



```

var
  P1, P2: PNode;

begin
  CreateStack(P1);
  CreateStack(P2);
  Cat(P1, P2);
  Writeln(P1^.Next);
end.

```

- 6) Дан указатель P_1 на вершину стека (если стек пуст, то $P_1 = \text{nil}$). Также дано число $N (> 0)$ и набор из N чисел. Описать тип TStack — запись с одним полем Top типа PNode (поле указывает на *вершину стека*) — и процедуру Push(S, D), которая добавляет в стек S новый элемент со значением D (S — входной и выходной параметр типа TStack, D — входной параметр целого типа). С помощью процедуры Push добавить в исходный стек данный набор чисел (последнее число будет вершиной стека) и вывести адрес новой вершины стека.

```

PascalABC.NET
Type
  PNode = ^TNode;
  TNode = record
    Data: Integer;
    Next: PNode;
  end;
  TStack = record
    Top: PNode;
  end;

procedure CreateStack(var P1: PNode);
var
  p: PNode;
  i, N: integer;

begin
  Write('N:');
  Readln(N);
  p1 := nil;
  for i := 1 to N do
  begin
    new(p);
    Write('num: ');
    Readln(P^.Data);
    p^.Next := P1;
    P1 := p;
  end;
end;

procedure Push(var S: TStack; D: Integer);
var
  temp: PNode;

```

```

begin
  new(temp);
  temp.Data := D;
  temp.Next := S.Top;
  S.Top := temp;
end;

var
  S: TStack;
  N, D, i: integer;

begin
  CreateStack(S.Top);
  Writeln('N:');
  Readln(N);
  for i := 1 to N do
  begin
    Writeln('num:');
    Readln(D);
    Push(S, D);
  end;
end.

```

2. Очереди

В заданиях структура «очередь» (queue) моделируется цепочкой связанных узлов-записей типа TNode. Поле Next последнего элемента цепочки равно nil. *Началом очереди* («головой», head) считается первый элемент цепочки, *концом* («хвостом», tail) — ее последний элемент. Для возможности быстрого добавления в конец очереди нового элемента удобно хранить, помимо указателя на начало очереди, также и указатель на ее конец. В случае пустой очереди указатели на ее начало и конец полагаются равными nil. Как и для стека, *значением* элемента очереди считается значение его поля Data.

7) Дан набор из 10 чисел. Создать очередь, содержащую данные числа в указанном порядке (первое число будет размещаться в начале очереди, последнее — в конце), и вывести указатели P_1 и P_2 на начало и конец очереди.

PascalABC.NET
<pre> Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; end; procedure CreateQueue(var P1, P2: PNode); var pt1, pt2: PNode; i, N: integer; begin Write('N:'); </pre>

```

Readln(N);
new(p1);
Write('num: ');
Readln(P1^.Data);
pt2 := p1;

for i := 1 to N - 1 do
begin
  new(pt1);
  Write('num: ');
  Readln(pt1^.Data);
  pt2^.Next := pt1;
  pt2 := pt2^.Next;
end;
pt2^.Next := nil;
p2 := pt2;
end;

var
  P1, P2: PNode;

begin
  CreateQueue(P1, P2);
  Writeln('adres:', @P1^);
  Writeln('adres:', @P2^);
end.

```

C++

```

#include <iostream>
using namespace std;
struct Node
{
    int data;
    Node *next;
};
struct Queue{
    int size;
    Node *first;
    Node *last;
};
void Creation(Queue *Q)
{
    Q->first = new Node;
    Q->first->next = NULL;
    Q->last = Q->first;
    Q->size = 0;
}
bool Full(Queue *Q)
{
    if (Q->first == Q->last)
        return true;
}

```

```

        else return false;
    }
    int Top(Queue *Q)
    {
        return Q->first->next->data;
    }
    void Add(Queue *Q)
    {
        int value;
        cout << "\nЗначение > "; cin >> value;
        Q->last->next = new Node;
        Q->last = Q->last->next;
        Q->last->data = value;
        Q->last->next = NULL;
        Q->size++;
    }
    void Delete(Queue *Q)
    {
        Q->first = Q->first->next;
        Q->size--;
        cout << "\nЭлемент удален\n";
    }
    int Size(Queue *Q)
    {
        return Q->size;
    }

    void main(){
        setlocale(LC_ALL, "Rus");
        Queue Q;
        Creation(&Q);
        for (int i = 0; i < 10; i++) {
            Add(&Q);
            if (i == 0)
                cout << &Q<<endl;
            if (i == 9)
                cout << &Q << endl;
        }
        system("pause");
    }
}

```

- 8) Дано число D и указатели P_1 и P_2 на начало и конец очереди (если очередь является пустой, то $P_1 = P_2 = \text{nil}$). Добавить элемент со значением D в конец очереди и вывести новые адреса начала и конца очереди.

PascalABC.NET	
Type	
	PNode = ^TNode;
	TNode = record
	Data: Integer;
	Next: PNode;

```

end;

procedure CreateQueue(var P1, P2: PNode);
var
    pt1, pt2: PNode;
    i, N: integer;
begin
    Write('N:');
    Readln(N);
    new(p1);
    Write('num: ');
    Readln(P1^.Data);
    pt2 := p1;

    for i := 1 to N - 1 do
        begin
            new(pt1);
            Write('num: ');
            Readln(pt1^.Data);
            pt2^.Next := pt1;
            pt2 := pt2^.Next;
        end;
        pt2^.Next := nil;
        p2 := pt2;
    end;

procedure Add(var P1, P2: PNode; D: integer);
var
    p: PNode;
begin
    new(p);
    p^.data := D;
    p^.next := nil;
    if P2 <> nil then
        begin
            P2^.Next := P;
            P2 := P2^.Next;
        end
    else
        begin
            P1 := p;
            P2 := p;
        end;
    end;

var
    P1, P2: PNode;
    a: ^Word;
    D: integer;

```

```

begin
  CreateQueue(P1, P2);

  Write('D:');
  Readln(D);
  Add(P1, P2, D);
  Writeln('P1: ', P1);
  Writeln('P2: ', P2);
end.

```

9) Даны указатели P_1 и P_2 на начало и конец непустой очереди. Извлекать из очереди элементы, пока значение начального элемента очереди не станет четным, и выводить значения извлеченных элементов (если очередь не содержит элементов с четными значениями, то извлечь все ее элементы). Вывести также новые адреса начала и конца очереди (для пустой очереди дважды вывести nil). После извлечения элементов из очереди освобождать память, которую они занимали.

PascalABC.NET
<pre> Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; end; procedure CreateQueue(var P1, P2: PNode); var pt1, pt2: PNode; i, N: integer; begin Write('NumEl:'); Readln(N); new(p1); Write('num: '); Readln(P1^.Data); pt2 := p1; for i := 1 to N - 1 do begin new(pt1); Write('num: '); Readln(pt1^.Data); pt2^.Next := pt1; pt2 := pt2^.Next; end; pt2^.Next := nil; p2 := pt2; end; function pop(var P1, P2: PNode): integer; var </pre>

```

    head: PNode;
begin
  if P1 <> nil then
    begin
      if P1 = P2 then
        begin
          Result := P1^.Data;
          P1 := nil;
          P2 := nil;
        end
      else
        begin
          Result := P1^.Data;
          head := P1^.Next;
          Dispose(P1);
          P1 := head;
        end;
      end
    end
  else Result := 0;
end;

procedure Add(var P1, P2: PNode; D: integer);
var
  p: PNode;
begin
  new(p);
  p^.data := D;
  p^.next := nil;
  if P2 <> nil then
    begin
      P2^.Next := P;
      P2 := P2^.Next;
    end
  else
    begin
      P1 := p;
      P2 := p;
    end;
end;

var
  P1, P2: PNode;
  a: ^Word;
  i, N: integer;

begin
  CreateQueue(P1, P2);
  while not (P1 = nil) do
    if not (odd(P1^.Data)) then break
    else Writeln(pop(P1, P2));

```

```

if p1 = nil then Writeln('nil')
else
  Writeln('odd_begin:', @P1
  );
if p2 = nil then Writeln('nil')
else
  Writeln('odd_end:', @P2^);
end.

```

3. Двусвязные списки

10) Дан адрес P_2 записи типа TNode, содержащей поле Data (целого типа) и поля Prev и Next (типа PNode — указателя на TNode). Эта запись связана полями Prev и Next соответственно с предыдущей и последующей записью того же типа. Вывести значения полей Data предыдущей и последующей записи, а также адреса P_1 и P_3 предыдущей и последующей записи.

PascalABC.NET
<pre> type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; Prev: PNode; end; var P1, P2, P3: PNode; D, i: integer; begin new(p2); new(p1); new(p3); Write('p1.Data: '); Readln(p1^.Data); Write('p3.Data: '); Readln(p3^.Data); P2^.Prev := P1; P2^.Next := P3; Writeln('P1.Data: ', p2^.prev^.Data); Writeln('P3.Data: ', p2^.next^.Data); Writeln('P1: ', @P2^.prev); Writeln('P3: ', @P2^.next); end. </pre>
C++
<pre> #include <iostream> using namespace std; struct TNode { int Data; TNode *next; </pre>


```

        TNode *prev;
    };
void main(){
    TNode *P1 = new TNode;
    TNode *P2 = new TNode;
    TNode *P3 = new TNode;
    int D, i;
    cout << "p1.Data: "; cin >> P1->Data;
    cout << "p3.Data: "; cin >> P3->Data;
    P2->prev = P1;
    P2->next = P3;
    cout << P2->prev->Data<<endl;
    cout << P2->next->Data << endl;
    cout << "P1=" << P2->prev << endl;
    cout << "P3=" << P2->next << endl;
    system("pause");
}

```

В заданиях структура «двусвязный список» (doubly linked list) моделируется цепочкой узлов-записей типа TNode, связанных как с предыдущим, так и с последующим узлом. Поле Next последнего элемента цепочки и поле Prev первого элемента цепочки равны nil. Для доступа к любому элементу двусвязного списка достаточно иметь указатель на один из его элементов, однако для ускорения операций со списком обычно хранят три указателя: на *первый* элемент списка (first), на его *последний* элемент (last) и на *текущий* элемент (current). Для пустого списка все эти указатели полагаются равными nil. Как в случае стека и очереди, *значением* элемента списка считается значение его поля Data.

11) Дан указатель P_1 на начало непустой цепочки элементов-записей типа TNode, связанных между собой с помощью поля Next. Используя поле Prev записи TNode, преобразовать исходную (*односвязную*) цепочку в *двусвязную*, в которой каждый элемент связан не только с последующим элементом (с помощью поля Next), но и с предыдущим (с помощью поля Prev). Поле Prev первого элемента положить равным nil. Вывести указатель на последний элемент преобразованной цепочки.

PascalABC.NET	
Type	PNode = ^TNode; TNode = record Data: Integer; Next: PNode; Prev: PNode; end;
procedure CreateStack(var P1: PNode);	
var	p, q: PNode; i, N: integer;
begin	Write('N:'); Readln(N);

```

New(P1);
Write('data: ');
Readln(P1^.Data);
P1^.Next := nil;
q := P1;
for i := 1 to N - 1 do
begin
  new(p);
  Write('data: ');
  Readln(P^.Data);
  p^.Next := nil;
  q^.Next := p;
  q := q^.Next
end;
end;

var
  P1, p, q: PNode;

begin
  CreateStack(P1);
  p := P1;
  p^.Prev := nil;
  while (p^.Next <> nil) do
  begin
    q := p;
    p := p^.Next;
    p^.Prev := Q;
  end;
  Writeln('end: ', @P);
end.

```

12) Дан указатель P_0 на один из элементов непустого двусвязного списка. Вывести число N — количество элементов в списке, а также указатели P_1 и P_2 на первый и последний элементы списка.

PascalABC.NET
<pre> Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; Prev: PNode; end; procedure CreateList(var P1, P2: PNode); var pt1, pt2: PNode; i, N: integer; begin Write('NumEl:'); Readln(N); </pre>

```

new(p1);
Write('num: ');
Readln(P1^.Data);
pt2 := p1;

for i := 1 to N - 1 do
begin
    new(pt1);
    Write('num: ');
    Readln(pt1^.Data);
    pt1^.Prev := pt2;
    pt2^.Next := pt1;
    pt2 := pt2^.Next;
end;
pt2^.Next := nil;
p2 := pt2;
end;

procedure InsertAfter(P0: PNode; D: integer);
var
    temp: PNode;
begin
    new(temp);
    temp^.Data := D;

    temp^.Prev := P0;
    temp^.Next := P0^.Next;

    P0^.Next := temp;

    if temp^.Next <> nil then
    begin
        temp := temp^.Next;
        temp^.Prev := P0^.Next;
    end;
end;

procedure DelNode(var P0, PPrev, PNext: PNode);
begin
    PNext := P0^.Next;
    PPrev := P0^.Prev;
    if PNext <> nil then PNext^.Prev := PPrev;
    if PPrev <> nil then PPrev^.Next := PNext;
    Dispose(P0);
    P0 := nil;
end;

var
    P0, P1, P2, PPrev, PNext: PNode;

begin

```

```

CreateList(P1, P2);
P0 := P1^.Next^.Next;
P0 := P1;
DelNode(P0, PPrev, PNext);
if PPrev = nil then
    Writeln('Prev: nil')
else
    Writeln('Prev: ', @PPrev);
if PNext = nil then
    Writeln('Next: nil')
else
    Writeln('Next: ', @PNext);
end.

```

- 13) Дан указатель P_1 на первый элемент непустого двусвязного списка. Продублировать в списке все элементы с четными значениями (новые элементы добавлять перед существующими элементами с такими же значениями) и вывести указатель на первый элемент преобразованного списка.

```

C++
#include<iostream>
#include<stdlib.h>
#include<time.h>
struct nodes {
    int data;
    nodes* prev;
    nodes* next;
};
nodes* generate(int n) {
    if (n<1) return NULL;
    srand(time(NULL));
    nodes* head = new nodes;
    head->next = NULL;
    head->prev = NULL;
    head->data = rand() % 10 + 1;
    for (int i = 1; i<n; i++) {
        head->prev = new nodes;
        head->prev->next = head;
        head->prev->prev = NULL;
        head->prev->data = rand() % 10 + 1;
        head = head->prev;
    }
    return head;
}
void show(nodes* ptr) {
    if (ptr == NULL) return;
    while (ptr != NULL) {
        std::cout << ptr->data << "\t";
        ptr = ptr->next;
    }
    std::cout << std::endl;
}
}

```

```

void destruct(nodes* ptr) {
    if (ptr == NULL) return;
    do {
        nodes* tmp = ptr;
        ptr = ptr->next;
        delete tmp;
    } while (ptr != NULL);
}
int main() {
    nodes* p1 = generate(5);
    show(p1);
    //=====
    nodes* iterator = p1;
    if (iterator->data % 2 == 0) {
        iterator->prev = new nodes;
        iterator->prev->data = iterator->data;
        iterator->prev->prev = NULL;
        iterator->prev->next = iterator;
        p1 = iterator->prev;
    }
    while (iterator->next != NULL) {
        if (iterator->next->data % 2 == 0) {
            iterator->next->prev = new nodes;
            iterator->next->prev->data = iterator->next->data;
            iterator->next->prev->next = iterator->next;
            iterator->next->prev->prev = iterator;
            iterator->next = iterator->next->prev;
            iterator = iterator->next;
        }
        iterator = iterator->next;
    }
    //=====
    show(p1);
    destruct(p1);
    p1 = NULL;
    system("pause");
    return 0;
}

```

- 14) Дан указатель P_0 на один из элементов непустого двусвязного списка. Переместить данный элемент в конец списка и вывести указатели на первый и последний элементы преобразованного списка. Операции выделения и освобождения памяти не использовать, поля Data не изменять.

PascalABC.NET	
Type	
	PNode = ^TNode;
	TNode = record
	Data: Integer;
	Next: PNode;

```

    Prev: PNode;
end;

procedure CreateList(var P1, P2: PNode);
var
    pt1, pt2: PNode;
    i, N: integer;
begin
    Write('NumEl:');
    Readln(N);
    new(p1);
    Write('num: ');
    Readln(P1^.Data);
    pt2 := p1;

    for i := 1 to N - 1 do
    begin
        new(pt1);
        Write('num: ');
        Readln(pt1^.Data);
        pt1^.Prev := pt2;
        pt2^.Next := pt1;
        pt2 := pt2^.Next;
    end;
    pt2^.Next := nil;
    p2 := pt2;
end;

function GetBeginPoint(P0: PNode): PNode;
begin
    Result := P0;
    while (Result^.Prev <> nil) do
        Result := Result^.Prev;
end;

function GetEndPoint(P0: PNode): PNode;
begin
    Result := P0;
    while (Result^.Next <> nil) do
        Result := Result^.Next;
end;

procedure MoveEnd(var P1, P2, P0: PNode);
begin
    P0^.Next^.Prev := P0^.Prev;
    P0^.Prev^.Next := P0^.Next;

    P0^.Next := nil;
    P0^.Prev := P2;

```

```

P2^.Next := P0;
P2 := P0;
end;

var
  P1, P2, P0: PNode;

begin
  CreateList(P1, P2);
  P0 := P1^.Next^.Next;
  P2 := nil;
  P1 := nil;
  P1 := GetBeginPoint(P0);
  P2 := GetEndPoint(P0);
  MoveEnd(P1, P2, P0);
  Writeln('begin: ', @P1);
  Writeln('end: ', @P2);
end.

```

- 15) Дан указатель P_1 на первый элемент непустого двусвязного списка. Перегруппировать его элементы, переместив все элементы с нечетными номерами в конец списка (в том же порядке) и вывести указатель на первый элемент преобразованного списка. Операции выделения и освобождения памяти не использовать, поля *Data* не изменять.

PascalABC.NET

```

Type
  PNode = ^TNode;
  TNode = record
    Data: Integer;
    Next: PNode;
    Prev: PNode;
  end;

procedure CreateList(var P1, P2: PNode);
var
  pt1, pt2: PNode;
  i, N: integer;
begin
  Write('NumEl:');
  Readln(N);
  new(p1);
  Write('num: ');
  Readln(P1^.Data);
  pt2 := p1;

  for i := 1 to N - 1 do
  begin
    new(pt1);
    Write('num: ');
    Readln(pt1^.Data);

```

```

    pt1^.Prev := pt2;
    pt2^.Next := pt1;
    pt2 := pt2^.Next;
end;
pt2^.Next := nil;
p2 := pt2;
end;

function GetEndPoint(P0: PNode): PNode;
begin
    Result := P0;
    while (Result^.Next <> nil) do
        Result := Result^.Next;
    end;
end;

procedure MoveEnd(var P1, P2: PNode; P0: PNode);
begin
    if P0 <> P2 then
    begin
        P0^.Next^.Prev := P0^.Prev;
        if P0^.Prev <> nil then P0^.Prev^.Next := P0^.Next
        else P1 := P0^.Next;
        P0^.Next := nil;
        P0^.Prev := P2;
        P2^.Next := P0;
        P2 := P0;
    end;
end;

procedure MoveOddEnd2(var P1, P2: PNode);
var
    p, q, pend: PNode;
    len: integer;
begin
    len := 0;
    p := P1;
    pend := P2;
    while (p <> pend) do
    begin
        inc(len);
        q := p;
        p := p^.Next;
        if odd(len) then MoveEnd(P1, P2, q);
    end;
    inc(len);
    if odd(len) then MoveEnd(P1, P2, pend);
end;
var

```



```

P1, P2: PNode;

begin
  CreateList(P1, P2);
  P2 := nil;
  P2 := GetEndPoint(P1);
  MoveOddEnd2(P1, P2);
  Writeln('begin: ', @P1);
end.

```

16) Дан указатель P_1 на первый элемент непустого двусвязного списка. Преобразовать список в *циклический*, связав его последний элемент с помощью поля Next с первым, а первый элемент с помощью поля Prev — с последним, и вывести указатель на элемент, который был последним элементом исходного списка.

PascalABC.NET
<pre> Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; Prev: PNode; end; procedure CreateList(var P1, P2: PNode); var pt1, pt2: PNode; i, N: integer; begin Write('NumEl:'); Readln(N); new(p1); Write('num: '); Readln(P1^.Data); pt2 := p1; for i := 1 to N - 1 do begin new(pt1); Write('num: '); Readln(pt1^.Data); pt1^.Prev := pt2; pt2^.Next := pt1; pt2 := pt2^.Next; end; pt2^.Next := nil; p2 := pt2; end; procedure cycle(var P1, P2: PNode); </pre>

```

begin
  P1^.Prev := P2;
  P2^.Next := P1;
end;
var
  P1, P2: PNode;
begin
  CreateList(P1, P2);
  P2 := nil;
  P2 := P1; //P2 := GetEndPoint(P1);
  cycle(P1, P2);
  Writeln('Old end: ', @P2);
end.

```

17) Даны указатели P_1 , P_2 и P_3 на первый, последний и текущий элементы двусвязного списка (если список является пустым, то $P_1 = P_2 = P_3 = \text{nil}$). Также дано число $N (> 0)$ и набор из N чисел. Описать тип TList — запись с полями First, Last и Current типа PNode (поля указывают соответственно на *первый*, *последний* и *текущий* элементы списка) — и процедуру InsertLast(L, D), которая добавляет новый элемент со значением D в конец списка L (L — входной и выходной параметр типа TList, D — входной параметр целого типа). Добавленный элемент становится текущим. С помощью этой процедуры добавить в конец исходного списка данный набор чисел (в том же порядке) и вывести новые адреса его первого, последнего и текущего элементов.

PascalABC.NET
<pre> Type PNode = ^TNode; TNode = record Data: Integer; Next: PNode; Prev: PNode; end; TList = record First, Last, Current: PNode; end; procedure InsertLast(var L: TList; D: integer); var temp: PNode; begin New(temp); temp^.Data := d; temp^.Next := nil; if L.First <> nil then begin temp^.Prev := L.Last; L.Last^.Next := temp; end else begin </pre>

```

    temp^.Prev := nil;
    L.First := temp;
end;
L.Last := temp;
L.Current := temp;
end;

var
  P1, P2, P3: PNode;
  L: TList;
  a: ^Word;
  N, D, i: integer;

begin
  P1 := nil;
  P2 := nil;
  P3 := nil;
  L.First := P1;
  L.Last := P2;
  L.Current := P3;
  //-----
  Write('N: ');
  Readln(N);
  for i := 1 to N do
    begin
      Write('D: ');
      Readln(D);
      InsertLast(L, D);
    end;
  Writeln('First: ', @L.First);
  Writeln('Last: ', @L.Last);
  Writeln('Current: ', @L.Current);
end.

```

Использованная в заданиях реализация двусвязного списка в виде цепочки узлов, ограниченной по краям нулевыми указателями, не является единственно возможной. Двусвязный список можно также реализовать в виде *замкнутой* цепочки узлов с дополнительным *фиктивным*, или *барьерным*, элементом. Этот барьерный элемент связан своими полями Next и Prev с первым и последним элементом списка соответственно, поэтому, имея указатель на барьерный элемент, можно сразу перейти как к первому, так и к последнему элементу списка (естественно, первый и последний элементы также связаны с барьерным элементом своими полями Prev и Next соответственно). Для работы с двусвязным списком, снабженным барьерным элементом, достаточно хранить два указателя: Barrier, указывающий на барьерный элемент, и Current, указывающий на текущий элемент (который может быть как «настоящим», так и барьерным элементом). Поле Data барьерного элемента может быть произвольным; для определенности будем полагать его равным 0. *Пустой список* в данной реализации представляет собой единственный барьерный элемент, «замкнутый на себя»; это означает, что для пустого списка поля Next и Prev

барьерного элемента равны *адресу барьерного элемента*, то есть значению указателя Barrier.

- 18) Даны указатели P_1 и P_2 на первый и последний элементы двусвязного списка, реализованного в виде цепочки узлов, ограниченной по краям нулевыми указателями (если список пуст, то $P_1 = P_2 = \text{nil}$). Преобразовать исходный список в *циклический список*, снабженный *барьерным элементом*. Барьерный элемент должен иметь значение 0 и быть связан своими полями Next и Prev с первым и последним элементом исходного списка (в случае пустого исходного списка поля Next и Prev барьерного элемента должны указывать на сам барьерный элемент). Вывести указатель на барьерный элемент полученного списка. Операцию выделения памяти использовать только для создания барьерного элемента.

```
PascalABC.NET
Type
  PNode = ^TNode;
  TNode = record
    Data: Integer;
    Next: PNode;
    Prev: PNode;
  end;

procedure CreateList(var P1, P2: PNode);
var
  pt1, pt2: PNode;
  i, N: integer;
begin
  Write('NumEl:');
  Readln(N);
  new(p1);
  Write('num: ');
  Readln(P1^.Data);
  pt2 := p1;

  for i := 1 to N - 1 do
  begin
    new(pt1);
    Write('num: ');
    Readln(pt1^.Data);
    pt1^.Prev := pt2;
    pt2^.Next := pt1;
    pt2 := pt2^.Next;
  end;
  pt2^.Next := nil;
  p2 := pt2;
end;

procedure ViewQueue(P1, P2: PNode);
var
  p: PNode;
  len: integer;
```

```

begin
  if P1 <> P2 then
    begin
      len := 0;
      p := p1;
      var a := @p;
      WriteLn(p^.Data);
      repeat
        p := p^.next;
        inc(len);
        WriteLn(p^.Data);
        if p = p2 then writeln('last address: ', a^);
      until (p = P2);

      writeln('len: ', len);
    end
  else Writeln('nil');
end;

function CycleList(var P1, P2: PNode): PNode;
var
  temp: PNode;
begin
  new(temp);
  temp^.Data := 0;
  temp^.Next := P1;
  temp^.Prev := P2;
  P1^.Prev := temp;
  P2^.Next := temp;
  CycleList := temp;
end;

var
  P1, P2, PB: PNode;
begin
  CreateList(P1, P2);
  PB := CycleList(P1, P2);
  Writeln('PB: ', @PB);
end.

```

- 19) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Также дано число $N (> 0)$ и набор из N чисел. Описать тип TListB — запись с полями Barrier и Current типа PNode (поля указывают соответственно на барьерный и текущий элементы списка) — и процедуру LBInsertLast(L , D), которая добавляет новый элемент со значением D в конец списка L (L — входной и выходной параметр типа TListB, D — входной параметр целого типа). Добавленный элемент становится текущим. С помощью этой процедуры добавить в конец исходного списка данный набор чисел (в том же порядке) и вывести адрес текущего элемента полученного списка.

```

Type
PNode = ^TNode;
TNode = record
    Data: Integer;
    Next: PNode;
    Prev: PNode;
end;
TListB = record
    Barrier, Current: PNode;
end;
procedure NewListB(var L: TListB);
begin
    new(L.Barrier);
    L.Barrier^.Data := 0;
    L.Barrier^.Next := L.Barrier;
    L.Barrier^.Prev := L.Barrier;
    L.Current := L.Barrier;
end;
procedure LBInsertLast(var L: TListB; D: integer);
var
    temp: PNode;
begin
    new(temp);
    temp^.Data := D;
    temp^.Next := L.Barrier;
    temp^.Prev := L.Barrier^.Prev;
    L.Barrier^.Prev^.Next := temp;
    L.Barrier^.Prev := temp;
    L.Current := temp;
end;

var
    P1, P2: PNode;
    N, D, i: integer;
    a: ^Word;
    L: TListB;
begin
    NewListB(L);
    P1 := L.Barrier;
    P2 := L.Current;
    //С этого момента, всё согласно заданию
    L.Barrier := P1;
    L.Current := P2;
    Write('N: ');
    Readln(N);
    for i := 1 to N do
    begin
        write('D: ');
        Readln(D);
        LBInsertLast(L, D);
    end;
    Writeln('current: ', @L.Current);
end.

```

Задачи для самостоятельной работы

Стеки

- 1) Дано число $N (> 0)$ и набор из N чисел. Создать стек, содержащий исходные числа (последнее число будет вершиной стека), и вывести указатель на его вершину.
- 2) Дан указатель P_1 на вершину непустого стека. Извлечь из стека первый (верхний) элемент и вывести его значение D , а также адрес P_2 новой вершины стека. Если после извлечения элемента стек окажется пустым, то положить $P_2 = \text{nil}$. После извлечения элемента из стека освободить память, занимаемую этим элементом.
- 3) Дан указатель P_1 на вершину стека (если стек пуст, то $P_1 = \text{nil}$). Извлечь из стека все элементы и вывести их значения. Вывести также количество извлеченных элементов N (для пустого стека вывести 0). После извлечения элементов из стека освобождать память, которую они занимали.
- 4) Даны указатели P_1 и P_2 на вершины двух непустых стеков. Перемещать элементы из первого стека во второй, пока значение вершины первого стека не станет четным (перемещенные элементы первого стека будут располагаться во втором стеке в порядке, обратном исходному). Если в первом стеке нет элементов с четными значениями, то переместить из первого стека во второй все элементы. Вывести адреса новых вершин первого и второго стека (если первый стек окажется пустым, то вывести для него константу nil). Операции выделения и освобождения памяти не использовать.
- 5) Дан указатель P_1 на вершину непустого стека. Создать два новых стека, переместив в первый из них все элементы исходного стека с четными значениями, а во второй — с нечетными (элементы в новых стеках будут располагаться в порядке, обратном исходному; один из этих стеков может оказаться пустым). Вывести адреса вершин полученных стеков (для пустого стека вывести nil). Операции выделения и освобождения памяти не использовать.
- 6) Дан указатель P_1 на вершину стека, содержащего не менее пяти элементов. Используя тип `TStack`, описать функцию `Pop(S)` целого типа, которая извлекает из стека S первый (верхний) элемент, возвращает его значение и освобождает память, которую занимал извлеченный элемент (S — входной и выходной параметр типа `TStack`). С помощью функции `Pop` извлечь из исходного стека пять элементов и вывести их значения. Вывести также указатель на новую вершину стека (если результирующий стек окажется пустым, то этот указатель должен быть равен nil).
- 7) Дан указатель P_1 на вершину стека. Используя тип `TStack`, описать функции `StackIsEmpty(S)` логического типа (возвращает `True`, если стек S пуст, и `False` в противном случае) и `Peek(S)` целого типа (возвращает значение вершины непустого стека S , не удаляя ее из стека). В обеих функциях переменная S является входным параметром типа `TStack`. С помощью этих функций, а также функции `Pop`, извлечь из исходного стека пять элементов (или все содержащиеся в нем элементы, если их менее пяти) и вывести их значения. Вывести также значение функции `StackIsEmpty` для результирующего стека и, если результирующий стек не является пустым, значение и адрес его новой вершины.

Очереди

- 8) Дан набор из 10 чисел. Создать две очереди: первая должна содержать числа из исходного набора с нечетными номерами (1, 3, ..., 9), а вторая — с четными (2, 4, ..., 10); порядок чисел в каждой очереди должен совпадать с порядком чисел в исходном наборе. Вывести указатели на начало и конец первой, а затем второй очереди.
- 9) Дан набор из 10 чисел. Создать две очереди: первая должна содержать все нечетные, а вторая — все четные числа из исходного набора (порядок чисел в каждой очереди должен совпадать с порядком чисел в исходном наборе). Вывести указатели на начало и конец первой, а затем второй очереди (одна из очередей может оказаться пустой; в этом случае вывести для нее две константы nil).
- 10) Дано число D и указатели P_1 и P_2 на начало и конец очереди, содержащей не менее двух элементов. Добавить элемент со значением D в конец очереди и извлечь из очереди первый (начальный) элемент. Вывести значение извлеченного элемента и новые адреса начала и конца очереди. После извлечения элемента из очереди освободить память, занимаемую этим элементом.
- 11) Дано число $N (> 0)$ и указатели P_1 и P_2 на начало и конец непустой очереди. Извлечь из очереди N начальных элементов и вывести их значения (если очередь содержит менее N элементов, то извлечь все ее элементы). Вывести также новые адреса начала и конца очереди (для пустой очереди дважды вывести nil). После извлечения элементов из очереди освобождать память, которую они занимали.
- 12) Даны две очереди; адреса начала и конца первой равны P_1 и P_2 , а второй — P_3 и P_4 (если очередь является пустой, то соответствующие адреса равны nil). Переместить все элементы первой очереди (в порядке от начала к концу) в конец второй очереди и вывести новые адреса начала и конца второй очереди. Операции выделения и освобождения памяти не использовать.
- 13) Дано число $N (> 0)$ и две непустые очереди; адреса начала и конца первой равны P_1 и P_2 , а второй — P_3 и P_4 . Переместить N начальных элементов первой очереди в конец второй очереди. Если первая очередь содержит менее N элементов, то переместить из первой очереди во вторую все элементы. Вывести новые адреса начала и конца первой, а затем второй очереди (для пустой очереди дважды вывести nil). Операции выделения и освобождения памяти не использовать.
- 14) Даны две непустые очереди; адреса начала и конца первой равны P_1 и P_2 , а второй — P_3 и P_4 . Перемещать элементы из начала первой очереди в конец второй, пока значение начального элемента первой очереди не станет четным (если первая очередь не содержит четных элементов, то переместить из первой очереди во вторую все элементы). Вывести новые адреса начала и конца первой, а затем второй очереди (для пустой очереди дважды вывести nil). Операции выделения и освобождения памяти не использовать.
- 15) Даны две непустые очереди; адреса начала и конца первой равны P_1 и P_2 , а второй — P_3 и P_4 . Очереди содержат одинаковое количество элементов. Объединить очереди в одну, в которой элементы исходных очередей чередуются (начиная с первого элемента первой очереди). Вывести указатели на начало и конец полученной очереди. Операции выделения и освобождения памяти не использовать.

- 16) Даны две непустые очереди; адреса начала и конца первой равны P_1 и P_2 , а второй — P_3 и P_4 . Элементы каждой из очередей упорядочены по возрастанию (в направлении от начала очереди к концу). Объединить очереди в одну с сохранением упорядоченности элементов. Вывести указатели на начало и конец полученной очереди. Операции выделения и освобождения памяти не использовать, поля Data не изменять.
- 17) Даны указатели P_1 и P_2 на начало и конец очереди (если очередь является пустой, то $P_1 = P_2 = \text{nil}$). Также дано число $N (> 0)$ и набор из N чисел. Описать тип TQueue — запись с двумя полями типа PNode: Head (*начало очереди*) и Tail (*конец очереди*) — и процедуру Enqueue(Q, D), которая добавляет в конец очереди Q новый элемент со значением D (Q — входной и выходной параметр типа TQueue, D — входной параметр целого типа). С помощью процедуры Enqueue добавить в исходную очередь данный набор чисел и вывести новые адреса ее начала и конца.
- 18) Даны указатели P_1 и P_2 на начало и конец очереди, содержащей не менее пяти элементов. Используя тип TQueue, описать функцию Dequeue(Q) целого типа, которая извлекает из очереди первый (начальный) элемент, возвращает его значение и освобождает память, занимаемую извлеченным элементом (Q — входной и выходной параметр типа TQueue). С помощью функции Dequeue извлечь из исходной очереди пять начальных элементов и вывести их значения. Вывести также адреса начала и конца результирующей очереди (если очередь окажется пустой, то эти адреса должны быть равны nil).
- 19) Даны указатели P_1 и P_2 на начало и конец очереди. Используя тип TQueue, описать функцию QueueIsEmpty(Q) логического типа, которая возвращает True, если очередь Q пуста, и False в противном случае (Q — входной параметр типа TQueue). Используя эту функцию для проверки состояния очереди, а также функцию Dequeue, извлечь из исходной очереди пять начальных элементов (или все содержащиеся в ней элементы, если их менее пяти) и вывести их значения. Вывести также значение функции QueueIsEmpty для полученной очереди и новые адреса ее начала и конца.

Двусвязные списки

- 20) Даны числа D_1 и D_2 и указатель P_0 на один из элементов непустого двусвязного списка. Добавить в начало списка новый элемент со значением D_1 , а в конец — новый элемент со значением D_2 . Вывести адреса первого и последнего элементов полученного списка.
- 21) Даны указатели P_1 и P_2 на первый и последний элементы двусвязного списка, содержащего не менее двух элементов. Продублировать в списке первый и последний элементы (новые элементы добавлять перед существующими элементами с такими же значениями) и вывести указатель на первый элемент преобразованного списка.
- 22) Дано число D и указатель P_0 на один из элементов непустого двусвязного списка. Вставить перед данным элементом списка новый элемент со значением D и вывести указатель на добавленный элемент списка.
- 23) Дано число D и указатель P_0 на один из элементов непустого двусвязного списка. Вставить после данного элемента списка новый элемент со значением D и вывести указатель на добавленный элемент списка.

- 24) Даны указатели P_1 и P_2 на первый и последний элементы двусвязного списка, содержащего не менее двух элементов. Продублировать в списке первый и последний элементы (новые элементы добавлять после существующих элементов с такими же значениями) и вывести указатель на последний элемент преобразованного списка.
- 25) Дан указатель P_1 на первый элемент непустого двусвязного списка. Продублировать в списке все элементы с нечетными номерами (новые элементы добавлять перед существующими элементами с такими же значениями) и вывести указатель на первый элемент преобразованного списка.
- 26) Дан указатель P_1 на первый элемент непустого двусвязного списка. Продублировать в списке все элементы с нечетными номерами (новые элементы добавлять после существующих элементов с такими же значениями) и вывести указатель на последний элемент преобразованного списка.
- 27) Дан указатель P_1 на первый элемент непустого двусвязного списка. Продублировать в списке все элементы с нечетными значениями (новые элементы добавлять после существующих элементов с такими же значениями) и вывести указатель на последний элемент преобразованного списка.
- 28) Дан указатель P_0 на один из элементов непустого двусвязного списка. Удалить из списка данный элемент и вывести два указателя: на элемент, предшествующий удаленному, и на элемент, следующий за удаленным (один или оба этих элемента могут отсутствовать; для отсутствующих элементов выводить nil). После удаления элемента из списка освободить память, занимаемую этим элементом.
- 29) Дан указатель P_1 на первый элемент двусвязного списка, содержащего не менее двух элементов. Удалить из списка все элементы с нечетными номерами и вывести указатель на первый элемент преобразованного списка. После удаления элементов из списка освобождать память, которую они занимали.
- 30) Дан указатель P_1 на первый элемент непустого двусвязного списка. Удалить из списка все элементы с нечетными значениями и вывести указатель на первый элемент преобразованного списка (если в результате удаления элементов список окажется пустым, то вывести nil). После удаления элементов из списка освобождать память, которую они занимали.
- 31) Дан указатель P_0 на один из элементов непустого двусвязного списка. Переместить данный элемент в начало списка и вывести указатели на первый и последний элементы преобразованного списка. Операции выделения и освобождения памяти не использовать, поля Data не изменять.
- 32) Дано число $K (> 0)$ и указатель P_0 на один из элементов непустого двусвязного списка. Переместить в списке данный элемент на K позиций вперед (если после данного элемента находится менее K элементов, то переместить его в конец списка). Вывести указатели на первый и последний элементы преобразованного списка. Операции выделения и освобождения памяти не использовать, поля Data не изменять.
- 33) Дано число $K (> 0)$ и указатель P_0 на один из элементов непустого двусвязного списка. Переместить в списке данный элемент на K позиций назад (если перед данным элементом находится менее K элементов, то переместить его в начало списка). Вывести указатели на первый и последний элементы преобразованного списка. Операции выделения и освобождения памяти не использовать, поля Data не изменять.

- 34) Даны указатели P_X и P_Y на два различных элемента двусвязного списка (элемент с адресом P_X находится в списке перед элементом с адресом P_Y , но не обязательно рядом с ним). Поменять местами данные элементы и вывести указатель на первый элемент преобразованного списка. Операции выделения и освобождения памяти не использовать, поля Data не изменять.
- 35) Дан указатель P_1 на первый элемент непустого двусвязного списка. Перегруппировать его элементы, переместив все элементы с нечетными значениями в конец списка (в том же порядке) и вывести указатель на первый элемент преобразованного списка. Операции выделения и освобождения памяти не использовать, поля Data не изменять.
- 36) Даны два непустых двусвязных списка и связанные с ними указатели: P_A и P_B указывают на первый и последний элементы первого списка, P_C — на один из элементов второго. Объединить исходные списки, поместив все элементы первого списка (в том же порядке) перед данным элементом второго списка, и вывести указатели на первый и последний элементы объединенного списка. Операции выделения и освобождения памяти не использовать.
- 37) Даны два непустых двусвязных списка и связанные с ними указатели: P_A и P_B указывают на первый и последний элементы первого списка, P_C — на один из элементов второго. Объединить исходные списки, поместив все элементы первого списка (в том же порядке) после данного элемента второго списка, и вывести указатели на первый и последний элементы объединенного списка. Операции выделения и освобождения памяти не использовать.
- 38) Даны указатели P_X и P_Y на два различных элемента двусвязного списка; элемент с адресом P_X находится в списке перед элементом с адресом P_Y , но не обязательно рядом с ним. Переместить элементы, расположенные между данными элементами (включая данные элементы) в новый список (в том же порядке). Вывести указатели на первые элементы преобразованного и нового списков. Если преобразованный список окажется пустым, то связанный с ним указатель положить равным nil. Операции выделения и освобождения памяти не использовать.
- 39) Даны указатели P_X и P_Y на два различных элемента двусвязного списка; элемент с адресом P_X находится в списке перед элементом с адресом P_Y , но не обязательно рядом с ним. Переместить элементы, расположенные между данными элементами (не включая данные элементы) в новый список (в том же порядке). Вывести указатели на первые элементы преобразованного и нового списков. Если новый список окажется пустым, то связанный с ним указатель положить равным nil. Операции выделения и освобождения памяти не использовать.
- 40) Даны указатели P_1 и P_2 на первый и последний элементы непустого двусвязного списка, содержащего четное количество элементов. Преобразовать список в два *циклических* списка, первый из которых содержит первую половину элементов исходного списка, а второй — вторую половину. Вывести указатели P_A и P_B на два средних элемента исходного списка (элемент с адресом P_A должен входить в первый циклический список, а элемент с адресом P_B — во второй). Операции выделения и освобождения памяти не использовать.
- 41) Дано число $K (> 0)$ и указатели P_1 и P_2 на первый и последний элементы непустого двусвязного списка. Осуществить *циклический сдвиг* элементов списка

на K позиций вперед (то есть в направлении от начала к концу списка) и вывести указатели на первый и последний элементы полученного списка. Для выполнения циклического сдвига преобразовать исходный список в циклический, после чего «разорвать» его в позиции, соответствующей данному значению K . Операции выделения и освобождения памяти не использовать.

42) Дано число $K (> 0)$ и указатели P_1 и P_2 на первый и последний элементы непустого двусвязного списка. Осуществить *циклический сдвиг* элементов списка на K позиций назад (то есть в направлении от конца к началу списка) и вывести указатели на первый и последний элементы полученного списка. Для выполнения циклического сдвига преобразовать исходный список в циклический, после чего «разорвать» его в позиции, соответствующей данному значению K . Операции выделения и освобождения памяти не использовать.

43) Даны указатели P_1 , P_2 и P_3 на первый, последний и текущий элементы двусвязного списка (если список является пустым, то $P_1 = P_2 = P_3 = \text{nil}$). Также дано число $N (> 0)$ и набор из N чисел. Используя тип TList, описать процедуру InsertFirst(L, D), которая добавляет новый элемент со значением D в начало списка L (L — входной и выходной параметр типа TList, D — входной параметр целого типа). Добавленный элемент становится текущим. С помощью этой процедуры добавить в начало исходного списка данный набор чисел (добавленные числа будут располагаться в списке в обратном порядке) и вывести новые адреса его первого, последнего и текущего элементов.

44) Дан непустой двусвязный список, первый, последний и текущий элементы которого имеют адреса P_1 , P_2 и P_3 . Также даны пять чисел. Используя тип TList, описать процедуру InsertBefore(L, D), которая вставляет новый элемент со значением D перед текущим элементом списка L (L — входной и выходной параметр типа TList, D — входной параметр целого типа). Вставленный элемент становится текущим. С помощью этой процедуры вставить пять данных чисел в исходный список и вывести новые адреса его первого, последнего и текущего элементов.

45) Дан непустой двусвязный список, первый, последний и текущий элементы которого имеют адреса P_1 , P_2 и P_3 . Также даны пять чисел. Используя тип TList, описать процедуру InsertAfter(L, D), которая вставляет новый элемент со значением D после текущего элемента списка L (L — входной и выходной параметр типа TList, D — входной параметр целого типа). Вставленный элемент становится текущим. С помощью этой процедуры вставить пять данных чисел в исходный список и вывести новые адреса его первого, последнего и текущего элементов.

46) Дан непустой двусвязный список, первый, последний и текущий элементы которого имеют адреса P_1 , P_2 и P_3 . Используя тип TList, описать процедуры ToFirst(L) (делает текущим первый элемент списка L), ToNext(L) (делает текущим в списке L следующий элемент, если он существует), SetData(L, D) (присваивает текущему элементу списка L значение D целого типа) и функцию IsLast(L) логического типа (возвращает True, если текущий элемент списка L является его последним элементом, и False в противном случае). Параметр L имеет тип TList; в процедурах ToFirst и ToNext он является входным и выходным. С помощью этих процедур и функций присвоить нулевые значения элементам исходного

списка с нечетными номерами и вывести количество элементов в списке, а также новый адрес текущего элемента списка.

- 47) Дан непустой двусвязный список, первый, последний и текущий элементы которого имеют адреса P_1 , P_2 и P_3 . Используя тип TList, описать процедуры ToLast(L) (делает текущим последний элемент списка L), ToPrev(L) (делает текущим в списке L предыдущий элемент, если он существует) и функции GetData(L) целого типа (возвращает значение текущего элемента списка L), IsFirst(L) логического типа (возвращает True, если текущий элемент списка L является его первым элементом, и False в противном случае). Параметр L имеет тип TList; в процедурах ToLast и ToPrev он является входным и выходным. С помощью этих процедур и функций вывести все четные значения элементов исходного списка, просматривая список с конца. Вывести также количество элементов в списке.
- 48) Даны указатели P_1 , P_2 и P_3 на первый, последний и текущий элементы двусвязного списка, содержащего не менее пяти элементов. Используя тип TList, описать функцию DeleteCurrent(L) целого типа, удаляющую из списка L текущий элемент и возвращающую его значение (L — входной и выходной параметр типа TList). После удаления элемента текущим становится следующий элемент или, если следующего элемента не существует, последний элемент списка. Функция также освобождает память, занимаемую удаленным элементом. С помощью этой функции удалить из исходного списка пять элементов и вывести их значения. Вывести также новые адреса первого, последнего и текущего элементов списка.
- 49) Даны указатели P_1 , P_2 и P_3 на первый, последний и текущий элементы непустого двусвязного списка. Используя тип TList, описать процедуру SplitList(L_1, L_2), которая переносит элементы списка L_1 от текущего до последнего в новый список L_2 (таким образом, список L_1 делится на две части, причем первая часть может оказаться пустой). Параметры процедуры имеют тип TList; первый параметр является входным и выходным, второй — выходным. Текущими элементами непустых результирующих списков становятся их первые элементы. Операции выделения и освобождения памяти в процедуре не использовать. С помощью этой процедуры разбить исходный список на два и вывести адреса первого, последнего и текущего элементов полученных списков.
- 50) Даны указатели на первый, последний и текущий элементы двух непустых двусвязных списков. Используя тип TList, описать процедуру AddList(L_1, L_2), которая добавляет все элементы из списка L_2 (в том же порядке) в конец списка L_1 ; в результате список L_2 становится пустым. Текущим элементом списка L_1 становится первый из добавленных элементов. Оба параметра процедуры имеют тип TList и являются входными и выходными. Операции выделения и освобождения памяти в процедуре не использовать. С помощью этой процедуры добавить второй из исходных списков в конец первого и вывести адреса первого, последнего и текущего элементов объединенного списка.
- 51) Даны указатели на первый, последний и текущий элементы двух непустых двусвязных списков. Используя тип TList, описать процедуру InsertList(L_1, L_2), которая вставляет все элементы из списка L_2 (в том же порядке) в список L_1 перед его текущим элементом; в результате список L_2 становится пустым. Текущим элементом списка L_1 становится первый из вставленных элементов. Оба параметра процедуры имеют тип TList и являются входными и выходными.

Операции выделения и освобождения памяти в процедуре не использовать. С помощью этой процедуры вставить второй из исходных списков в текущую позицию первого и вывести адреса первого, последнего и текущего элементов объединенного списка.

- 52) Даны указатели на первый, последний и текущий элементы двух двусвязных списков (второй список может быть пустым). Используя тип TList, описать процедуру MoveCurrent(L_1, L_2), которая перемещает текущий элемент списка L_1 в список L_2 (элемент вставляется после текущего элемента списка L_2 и сам становится текущим; в списке L_1 текущим становится следующий элемент или, если следующего элемента не существует, последний элемент). Оба параметра процедуры имеют тип TList и являются входными и выходными. Операции выделения и освобождения памяти в процедуре не использовать. С помощью этой процедуры переместить текущий элемент первого списка во второй и вывести адреса первого, последнего и текущего элементов полученных списков.
- 53) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Разбить список на два, перенести во второй список все элементы от текущего до последнего и добавив ко второму списку барьерный элемент. Если текущий элемент исходного списка является барьерным элементом, то второй список должен быть пустым (то есть состоять только из барьерного элемента). Вывести указатель на барьерный элемент второго списка. Операцию выделения памяти использовать только для создания барьерного элемента второго списка.
- 54) Даны указатели P_1 и P_2 на барьерные элементы двух двусвязных списков. Объединить исходные списки, связав конец первого и начало второго списка (барьерным элементом объединенного списка должен остаться барьерный элемент первого списка). Вывести указатели на первый и последний элементы объединенного списка (если объединенный список является пустым, то дважды вывести указатель на его барьерный элемент). После удаления лишнего барьерного элемента освободить занимаемую им память.
- 55) Даны указатели P_1 и P_2 на барьерные элементы двух двусвязных списков. Объединить исходные списки, связав конец первого и начало второго списка (барьерным элементом объединенного списка должен остаться барьерный элемент второго списка). Вывести указатели на первый и последний элементы объединенного списка (если объединенный список является пустым, то дважды вывести указатель на его барьерный элемент). После удаления лишнего барьерного элемента освободить занимаемую им память.
- 56) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Также дано число N (> 0) и набор из N чисел. Используя тип TListB, описать процедуру LBInsertFirst(L, D), которая добавляет новый элемент со значением D в начало списка L (L — входной и выходной параметр типа TListB, D — входной параметр целого типа). Добавленный элемент становится текущим. С помощью этой процедуры добавить в начало исходного списка данный набор чисел (добавленные числа будут располагаться в списке в обратном порядке) и вывести адрес текущего элемента полученного списка.
- 57) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Также даны пять чисел. Используя тип TListB, описать процедуру LBInsertBefore(L, D), которая вставляет новый элемент со значением D перед текущим элементом списка L (L — входной и выходной параметр типа TListB, D

— входной параметр целого типа). Вставленный элемент становится текущим. С помощью этой процедуры вставить пять данных чисел в исходный список и вывести новый адрес его текущего элемента.

- 58) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Также даны пять чисел. Используя тип `TListB`, описать процедуру `LBInsertAfter(L, D)`, которая вставляет новый элемент со значением D после текущего элемента списка L (L — входной и выходной параметр типа `TListB`, D — входной параметр целого типа). Вставленный элемент становится текущим. С помощью этой процедуры вставить пять данных чисел в исходный список и вывести новый адрес его текущего элемента.
- 59) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Используя тип `TListB`, описать процедуры `LBToFirst(L)` (делает текущим первый элемент списка L), `LBToNext(L)` (делает текущим в списке L следующий элемент), `LBSetData(L, D)` (присваивает текущему элементу списка L значение D целого типа, если данный элемент не является барьерным) и функцию `IsBarrier(L)` логического типа (возвращает `True`, если текущий элемент списка L является его барьерным элементом, и `False` в противном случае). Параметр L имеет тип `TListB`; в процедурах `LBToFirst` и `LBToNext` он является входным и выходным. С помощью этих процедур и функций присвоить нулевые значения элементам исходного списка с нечетными номерами и вывести количество элементов в списке, а также новый адрес текущего элемента списка. Барьерный элемент при подсчете элементов не учитывать.
- 60) Даны указатели P_1 и P_2 на барьерный и текущий элементы двусвязного списка. Используя тип `TListB`, описать процедуры `LBToLast(L)` (делает текущим последний элемент списка L), `LBToPrev(L)` (делает текущим в списке L предыдущий элемент) и функцию `LBGetData(L)` целого типа (возвращает значение текущего элемента списка L). Параметр L имеет тип `TListB`; в процедурах `LBToLast` и `LBToPrev` он является входным и выходным. С помощью этих процедур и функций, а также с использованием функции `IsBarrier` из задания `Pointer78`, вывести все четные значения элементов исходного списка, просматривая список с конца. Вывести также количество элементов в списке. Барьерный элемент при подсчете элементов не учитывать.
- 61) Даны указатели P_1 и P_2 на барьерный и текущий элементы непустого двусвязного списка, причем текущий элемент не совпадает с барьерным. Используя тип `TListB`, описать функцию `LBDeleteCurrent(L)` целого типа, удаляющую из списка L текущий элемент и возвращающую его значение (L — входной и выходной параметр типа `TListB`). Текущим становится следующий элемент или, если следующий элемент является барьерным, предыдущий элемент списка. Функция также освобождает память, занимаемую удаленным элементом. Если текущим элементом является барьерный элемент, то функция не выполняет никаких действий и возвращает 0. С помощью этой функции, а также функции `IsBarrier`, удалить из исходного списка пять элементов (или все элементы, если их менее пяти) и вывести их значения. Вывести также новый адрес текущего элемента списка.

Литература

- 1) Абрамов С. А., Зима Е. В. Начала информатики. – М.: Наука, 1989. – 256 с.
- 2) Абрамян М. Э. Практикум по программированию на языке Паскаль. 4-е изд. — Ростов-на-Дону: Изд-во «ЦВВР», 2004. — 187 с.
- 3) Абрамян М. Э., Михалкович С. С. Основы программирования на языке Паскаль: Скалярные типы данных, управляющие операторы, процедуры и функции. — Ростов-на-Дону: Изд-во «ЦВВР», 2004. — 198 с.
- 4) Абрамян М.Э. 1000 задач по программированию. Часть 1. Ростов-на Дону. 2004. - 154 с.
- 5) Арабов М.К., Нуров И.Дж. Основы программирования на языке PascalABC.NET. Учебно-методическое пособие. – Душанбе: РТСУ; 2018. - 280 с.
- 6) Арабов М.К, Гулов А.М, Нурализода А., Тимурбекова С. Маҷмӯи корҳои лабораторӣ аз ЗБСО. Душанбе. 2018. - 95 С.
- 7) Арабов М.Қ. Асосҳои барноманависӣ дар С++. Душанбе. РТСУ 2018. - 400с.
- 8) Габасов Р., Кириллова Ф. М. Основы динамического программирования. — Мн.: Изд-во БГУ, 1975. — 262 с.
- 9) Герберт Шилдт. С: полное руководство, классическое издание.— М.: Вильямс, 2010. — 704 с.
- 10) Герберт Шилдт. Полный справочник по С++ . — М.: Вильямс, 2011. — 800 с.
- 11) Голуб А. И. С и С++. Правила программирования. - М: БИНОМ, 1996 г. - 272 с.
- 12) Гукин Д. Язык программирования Си для «чайников».— М.: Диалектика, 2006. — 352 с.
- 13) Дейтел Х., Дейтел П. Как программировать на С++.— М.: БИНОМ, 1998 - 1024 с.
- 14) М. Арабов, М. Халилова. Маҷмӯи мисол ва масъалаҳо аз барномасозӣ. Душанбе. ДСРТ, 2018. - 138 с.
- 15) Муллошараф Арабов. Асосҳои барноманависии дар забони PascalABC.NET. Китоби дарсӣ. Душанбе. РТСУ - 2018. - 302с.
- 16) Шилдт Г. С# 4.0: полное руководство.: Пер. с англ. — М.: ООО "И.Д. Вильямс", 2011. — 1056 с.
- 17) Шилдт Г. Полный справочник по С++ . — М.: Вильямс, 2011.- 800 с.
- 18) Шилдт Г. Самоучитель С++: Пер. с англ. - Санкт-Петербург: ВHV-Санкт-Петербург, 1998. – 620 с.
- 19) <https://github.com/>
- 20) <https://pythonworld.ru/>
- 21) <http://golang-book.ru/>
- 22) <https://proglib.io/p/30-golang-books/>
- 23) <http://cppstudio.com>
- 24) <https://metanit.com/>

Технический редактор: Н. Тохиров
Компьютерная верстка: А. Тохиров

© Издательство РТСУ

Сдано в набор 26.02.2019. Подписано в печать 27.02.2019.
Бумага офсетная. Печать офсетная. Гарнитура литературная.
Формат 60x84_{1/16}. Услов. печ. л. 23.
Тираж 100 экз. Заказ № 503.

Отпечатано в типографии РТСУ,
734025, Республика Таджикистан, г. Душанбе,
ул. Мирзо Турсун-заде-30

