

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное учреждение  
высшего профессионального образования  
"Казанский (Приволжский) федеральный университет"  
Институт вычислительной математики и информационных технологий



УТВЕРЖДАЮ

Проректор по образовательной деятельности КФУ

Проф. Тагорский Д.А.



\_\_\_\_\_ 20\_\_ г.

*подписано электронно-цифровой подписью*

**Программа дисциплины**  
Программирование Б3.Б.4

Направление подготовки: 080500.62 - Бизнес-информатика

Профиль подготовки:

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

**Автор(ы):**

Фофанов В.Б.

**Рецензент(ы):**

Тагиров Р.Р. , Миссаров М.Д.

**СОГЛАСОВАНО:**

Заведующий(ая) кафедрой: Миссаров М. Д.

Протокол заседания кафедры No \_\_\_\_ от " \_\_\_\_ " \_\_\_\_\_ 201\_\_ г

Учебно-методическая комиссия Института вычислительной математики и информационных технологий:

Протокол заседания УМК No \_\_\_\_ от " \_\_\_\_ " \_\_\_\_\_ 201\_\_ г

Регистрационный No 938616

Казань  
2016

## Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре основной образовательной программы
3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля
4. Структура и содержание дисциплины/ модуля
5. Образовательные технологии, включая интерактивные формы обучения
6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов
7. Литература
8. Интернет-ресурсы
9. Материально-техническое обеспечение дисциплины/модуля согласно утвержденному учебному плану

Программу дисциплины разработал(а)(и) Фофанов В.Б. , Viatcheslav.Fofanov@kpfu.ru

### 1. Цели освоения дисциплины

Данная учебная дисциплина реализуется как цикл лекционных и лабораторных занятий, которые знакомят студентов с основами программирования и формируют у них навыки разработки программ на языках программирования высокого уровня. В качестве основного языка программирования, который будет изучаться в данном курсе, выбран популярный сейчас язык программирования C#.

В рамках данного курса студенты должны получить представление об основных алгоритмических конструкциях, на основе которых строятся программы, основных синтаксических конструкциях языка программирования C# (типы данных, объявление переменных, операции языка, операторы). Также студенты должны получить навыки разработки программ в стиле структурного программирования, разделяя алгоритм на пользовательские процедуры и функции. Отдельно подлежат разбору классические алгоритмы решения разнообразных задач информатики, а также принципы построения и использования для решения задач различных видов динамических структур данных, таких как линейные односвязные и двусвязные списки, стеки и очереди, графы и бинарные деревья сортировки.

### 2. Место дисциплины в структуре основной образовательной программы высшего профессионального образования

Данная учебная дисциплина включена в раздел " Б3.Б.4 Профессиональный" основной образовательной программы 080500.62 Бизнес-информатика и относится к базовой (общепрофессиональной) части. Осваивается на 1 курсе, 1 семестр.

Дисциплина "Программирование" изучается на 1 курсе в 1 семестре обучения бакалавров. Знания по этому курсу требуются при изучении учебных дисциплин "Алгоритмы и структуры данных", "Объектно-ориентированный анализ и программирование", "Интернет-технологии", "Вычислительные системы и телекоммуникации", "Проектирование информационных систем" и других учебных дисциплин, которые предусмотрены учебным планом по направлению "Бизнес-информатика". Навыки, полученные при изучении этого предмета, будут использованы студентами при написании курсовых и дипломных работ.

### 3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОК-12 (общекультурные компетенции)	осознает сущность и значение информации в развитии современного общества; владеет основными методами, способами и средствами получения, хранения, переработки информации;
ОК-16 (общекультурные компетенции)	способен работать с информацией из различных источников;
ПК-20 (профессиональные компетенции)	использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования;
ПК-21 (профессиональные компетенции)	отовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований

В результате освоения дисциплины студент:

## 1. должен знать:

- основные понятия алгоритмизации;
- синтаксис и семантику языка программирования C#;
- принципы структурного программирования;
- базовые алгоритмы решения стандартных задач (поиска, сортировки и пр.).

## 2. должен уметь:

- представить алгоритм в виде блок-схемы и записать его на языке программирования C#;
- иметь навыки оформления программы в стиле структурного программирования в виде набора пользовательских функций;
- уметь применять основные алгоритмы теории сортировки, поиска, нахождения наилучших решений;

## 3. должен владеть:

- навыками разработки алгоритмов решения вычислительных задач;
- навыками разработки консольных приложений в среде разработки семейства Visual Studio.

- применять полученные знания и навыки при изучении следующих по учебному плану дисциплин, при написании курсовых и дипломной работы и в своей дальнейшей деятельности

**4. Структура и содержание дисциплины/ модуля**

Общая трудоемкость дисциплины составляет 5 зачетных(ые) единиц(ы) 180 часа(ов).

Форма промежуточного контроля дисциплины экзамен в 1 семестре.

Суммарно по дисциплине можно получить 100 баллов, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов. Минимальное количество для допуска к зачету 28 баллов.

86 баллов и более - "отлично" (отл.);

71-85 баллов - "хорошо" (хор.);

55-70 баллов - "удовлетворительно" (удов.);

54 балла и менее - "неудовлетворительно" (неуд.).

**4.1 Структура и содержание аудиторной работы по дисциплине/ модулю****Тематический план дисциплины/модуля**

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. 1. Основные типы данных	1	1	1	0	5	домашнее задание
2.	Тема 2. 2. Операторы	1	2-3	2	0	3	домашнее задание
3.	Тема 3. 3. Функции. Начало	1	4	1	0	5	домашнее задание

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
4.	Тема 4. 4. Производные типы	1	5-6	5	0	5	домашнее задание
5.	Тема 5. 5. Функции. Продолжение	1	7-8	4	0	5	домашнее задание
6.	Тема 6. 6. Структуры, перечисления, списки	1	9-11	4	0	6	контрольная работа домашнее задание
7.	Тема 7. Обзор некоторых функций стандартных библиотек. 7. Файлы и их применение	1	12-14	4	0	7	домашнее задание
8.	Тема 8. Структурное программирование	1	15-16	4	0	5	домашнее задание
9.	Тема 9. Рекурсивные функции.	1	17-18	3	0	5	контрольная работа домашнее задание
	Тема . Итоговая форма контроля	1		0	0	0	экзамен
	Итого			28	0	46	

## 4.2 Содержание дисциплины

### Тема 1. 1. Основные типы данных

#### **лекционное занятие (1 часа(ов)):**

История создания языков C и C++. Алфавит языка, синтаксис и семантика. Имя (идентификатор) объекта. Правила идентификации объектов.

#### **лабораторная работа (5 часа(ов)):**

Изучение алфавита языка, синтаксиса и семантикаи. Имя (идентификатор) объекта. Правила идентификации объектов.

### Тема 2. 2. Операторы

#### **лекционное занятие (2 часа(ов)):**

Общие понятия, связанные с типами данных. Характеристики логического типа, символьного типа, целых типов, типов с плавающей точкой, перечислимых типов, типа void, массивов, ссылок, структур данных. Структуры данных как тип данных, определяемый пользователем. Размеры объектов.

#### **лабораторная работа (3 часа(ов)):**

Примеры разных типов данных

### Тема 3. 3. Функции. Начало

#### **лекционное занятие (1 часа(ов)):**

Объявления и определения объектов. Синтаксис объявления объектов разных типов данных. Понятие области видимости идентификатора. Локальные и глобальные переменные.

**лабораторная работа (5 часа(ов)):**

Задания на объявление данных разных типов. Рассмотрение локальных и глобальных переменных.

**Тема 4. 4. Производные типы**

**лекционное занятие (5 часа(ов)):**

Виды операций языка C#. Неявное преобразование типов. Классификация операций. Правила построения выражений. Приоритеты операций.

**лабораторная работа (5 часа(ов)):**

Рассмотрение операций языка C#. Примеры неявного преобразования типов. Приоритеты операций.

**Тема 5. 5. Функции. Продолжение**

**лекционное занятие (4 часа(ов)):**

Виды операторов: объявления, составной оператор, выражения, условный оператор, оператор выбора, операторы цикла, операторы передачи управления.

**лабораторная работа (5 часа(ов)):**

Рассмотрение на примерах разных видов операторов: объявления, составной оператор, выражения, условный оператор, оператор выбора, операторы цикла, операторы передачи управления.

**Тема 6. 6. Структуры, перечисления, списки**

**лекционное занятие (4 часа(ов)):**

Примеры работы с циклами, массивами, структурами, матрицами. Два способа хранения матрицы.

**лабораторная работа (6 часа(ов)):**

Написание фрагментов программ с использованием циклов, массивов, структур, матриц. Два способа хранения матрицы.

**Тема 7. Обзор некоторых функций стандартных библиотек. 7. Файлы и их применение**

**лекционное занятие (4 часа(ов)):**

Понятие стандартной библиотеки. Обзор библиотечных "обычных математических функций". Работа со случайными числами. Обзор библиотечных функций для работы с символьными строками. Обзор библиотечных функций для работы с файлами. Примеры.

**лабораторная работа (7 часа(ов)):**

Примеры работы с функциями функций стандартных библиотек.

**Тема 8. Структурное программирование**

**лекционное занятие (4 часа(ов)):**

Структурное программирование. Проектирование сверху вниз и снизу вверх. Синтаксис объявления и определения функций. Синтаксис вызова функции. Формальные и фактические параметры. Способы передачи параметров в функцию. Особенности передачи в функцию параметров-массивов. Делегаты. Примеры.

**лабораторная работа (5 часа(ов)):**

Рассмотрение функций: структурное программирование. Проектирование сверху вниз и снизу вверх. Примеры на объявление и определение функций. Синтаксис вызова функции. Формальные и фактические параметры. Примеры способов передачи параметров в функцию. Особенности передачи в функцию параметров-массивов.

**Тема 9. Рекурсивные функции.**

**лекционное занятие (3 часа(ов)):**

Определение рекурсии. Случаи использования. Примеры - вычисление  $n!$ , числа Фибоначчи, задача о Ханойской башне, алгоритм быстрой сортировки.

**лабораторная работа (5 часа(ов)):**

Рассмотрение рекурсивных функций. Решение примеров: вычисление  $n!$ , числа Фибоначчи, задача о Ханойской башне, алгоритм быстрой сортировки.

#### 4.3 Структура и содержание самостоятельной работы дисциплины (модуля)

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
1.	Тема 1. 1. Основные типы данных	1	1	подготовка домашнего задания	6	домашнее задание
2.	Тема 2. 2. Операторы	1	2-3	подготовка домашнего задания	8	домашнее задание
3.	Тема 3. 3. Функции. Начало	1	4	подготовка домашнего задания	8	домашнее задание
4.	Тема 4. 4. Производные типы	1	5-6	подготовка домашнего задания	8	домашнее задание
5.	Тема 5. 5. Функции. Продолжение	1	7-8	подготовка домашнего задания	8	домашнее задание
6.	Тема 6. 6. Структуры, перечисления, списки	1	9-11	подготовка домашнего задания	6	домашнее задание
				подготовка к контрольной работе	2	контрольная работа
7.	Тема 7. Обзор некоторых функций стандартных библиотек. 7. Файлы и их применение	1	12-14	подготовка домашнего задания	8	домашнее задание
8.	Тема 8. Структурное программирование	1	15-16	подготовка домашнего задания	8	домашнее задание
9.	Тема 9. Рекурсивные функции.	1	17-18	подготовка домашнего задания	6	домашнее задание
				подготовка к контрольной работе	2	контрольная работа
Итого					70	

#### 5. Образовательные технологии, включая интерактивные формы обучения

Дисциплина представляет собой цикл лекционных и лабораторных занятий. Лабораторные занятия посвящены выработке навыков разработки программ на языке программирования C# для решения различных вычислительных задач. Лабораторные занятия проходят в компьютерных классах с использованием интерактивной доски для наглядного представления алгоритмов и разработки программ на всех этапах ее создания и компиляции. Лабораторные занятия проходят в интерактивной форме обсуждения решения различных задач или в активной форме самостоятельного решения задач студентами. Контроль за выполнением самостоятельной работы проявляется в функциональном тестировании выполненных студентами заданий на примерах, предложенных преподавателем.

## **6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов**

### **Тема 1. 1. Основные типы данных**

домашнее задание , примерные вопросы:

1. Привести примеры объектов типа `int` и выполнить с ними операции сравнения, логические операции, операции присваивания, побитовые операции. 2. Привести примеры объектов типа `float` и выполнить с ними операции сравнения, логические операции, операции присваивания, побитовые операции. 3. Привести примеры объектов типа `char` и выполнить с ними операции сравнения, логические операции, операции присваивания, побитовые операции.

### **Тема 2. 2. Операторы**

домашнее задание , примерные вопросы:

1. Написать несколько составных операторов. 2. Привести несколько примеров условных операторов и безусловных операторов. 3. Привести несколько примеров операторов цикла. 4. Привести пример оператора многоальтернативного выбора.

### **Тема 3. 3. Функции. Начало**

домашнее задание , примерные вопросы:

1. Привести примеры функций. 2. Привести примеры стандартных функций. 3. Написать фрагмент программы вызова функции с указанием фактических параметров.

### **Тема 4. 4. Производные типы**

домашнее задание , примерные вопросы:

1. Привести пример с адресом объекта в качестве параметра функции. 2. Объяснить назначение функции `Change()`. 3. Объяснить назначение функций `printf()` и `scanf()`.

### **Тема 5. 5. Функции. Продолжение**

домашнее задание , примерные вопросы:

1. Привести примеры функций с локальными и глобальными идентификаторами. 2. Привести примеры сокрытия глобальных идентификаторов. 3. Привести примеры с автоматическими и статическими идентификаторами.

### **Тема 6. 6. Структуры, перечисления, списки**

домашнее задание , примерные вопросы:

1. Привести примеры массивов. 2. Привести примеры определения адреса массива и адреса элемента массива. 3. Привести примеры передачи массива в функцию. 4. Привести примеры объединения массивов с одинаковой длиной и с разной длиной. 5. Написать примеры создания массивов из элементов таблицы и из строк таблицы. 6. Привести примеры использования функции `rand()`.

контрольная работа , примерные вопросы:

Подготовка к контрольной работе (выполнению индивидуальных заданий) по всем пройденным темам

### **Тема 7. Обзор некоторых функций стандартных библиотек. 7. Файлы и их применение**

домашнее задание , примерные вопросы:



1. Привести примеры библиотечных функций для работы с символьными строками. 2. Привести примеры библиотечных функций для работы с файлами. 3. Привести примеры файлов разных типов (docx, xlsx, pdf, jpg, bat, com, exe и др.). 4. Написать операторы открытия и закрытия файлов. 5. Написать операторы ввода информации из файла и вывода информации в файл. 1.

### **Тема 8. Структурное программирование**

домашнее задание , примерные вопросы:

1. Написать фрагмент программы с объявлением и вызовом функции. 1. Привести примеры способов передачи параметров в функцию. 2. Привести примеры передачи в функцию параметров-массивов.

### **Тема 9. Рекурсивные функции.**

домашнее задание , примерные вопросы:

1. Написать программу для вычисления  $n!$  с использованием рекурсивной функции. 2. Написать программу для вычисления числа Фибоначчи с использованием рекурсивной функции. 3. Написать программу для решения задачи о Ханойской башне с использованием рекурсивной функции.

контрольная работа , примерные вопросы:

Подготовка к контрольной работе (выполнению индивидуальных заданий) по теме "Пользовательские и библиотечные функции".

### **Тема . Итоговая форма контроля**

Примерные вопросы к экзамену:

По данной дисциплине предусмотрено проведение экзамена.

Билеты к экзамену:

**БИЛЕТ 1**

1. Структура программы и основные конструкции
2. Написать программу для вычисления синуса с заданной точностью для заданного  $x$  по формуле
3. Написать программу, которая в текстовом файле заменит все вхождения одного заданного слова на другое заданное слово

**БИЛЕТ 2**

1. Операторы цикла
2. Написать программу для вычисления косинуса с заданной точностью для заданного  $x$  по формуле
3. Написать программу для вычисления коэффициентов  $p_0, p_1, p_2, \dots, p_n$  многочлена степени  $n$  по заданным корням  $x_1, x_2, \dots, x_n$

**БИЛЕТ 3**

1. Простейший ввод-вывод (клавиатура + экран)
2. Написать программу для вычисления экспоненты с заданной точностью для заданного  $x$  по формуле
3. Написать программу для печати всех слов из заданного текста (символьной строки) в обратном порядке (от последнего к первому)

**БИЛЕТ 4**

1. Массивы
2. Написать программу для обхода дерева с рекурсией
3. В тексте (символьной строке) находятся целые числа без знака в символьном виде. Написать программу, которая найдет и напечатает их сумму.

**БИЛЕТ 5**

1. Символьные строки
2. Написать программу для нахождения минимума и максимума последовательности целых

чисел без массивов

3. Написать программу, которая напечатает линейный список в обратном порядке

БИЛЕТ 6

1. Указатели, адреса и адресная арифметика

2. Написать программу для нахождения минимума и среднего арифметического максимума последовательности целых чисел без массивов

3. Написать программу, которая в тексте (символьной строке) со-считает количество слов, оканчивающихся на "ing"

БИЛЕТ 7

1. Сортировки и поиск, слияние

1. Написать программу для вычисления факториала с рекурсией и без

3. Написать программу, которая в квадратной матрице вычислит сумму элементов по всем диагоналям одного направления

БИЛЕТ 8

1. Область видимости и время жизни объектов

2. Написать программу для вычисления чисел Фибоначчи с рекурсией и без

3. Написать программу, которая в целочисленном массиве найдет наибольшую сумму подряд идущих элементов.

БИЛЕТ 9

1. Модульное программирование

2. Написать программу для упорядочения массива целых чисел путем сравнением соседних ("пузырьком")

3. Написать программу, которая напечатает из текста (символьной строки) все слова по алфавиту

БИЛЕТ 10

1. Структуры

2. Дана символьная строка. Написать программу для печати каждого слова с новой строки и его длины

3. Написать программу, которая в квадратной матрице вычислит сумму элементов, которые находятся или под главной или под побочной диагональю

БИЛЕТ 11

1. Линейные списки

2. Написать программу для упорядочения массива целых чисел путем нахождения минимума

3. Написать программу, которая в тексте найдет и напечатает все слова, которые начинаются с заглавной буквы и не содержат ничего, кроме латинских букв

БИЛЕТ 12

1. Рекурсия

2. Написать программу для вычисления значения многочлена для заданного значения аргумента  $x$

3. Написать программу для нахождения разности двух упорядоченных массивов целых чисел

БИЛЕТ 13

1. Стек, очередь, ДЭК

2. Дана символьная строка. Написать программу для печати самого длинного слова

3. Написать программу, которая в прямоугольной матрице вычислит сумму элементов, которые находятся на одной вертикали, горизонтали или диагонали с заданным элементом

БИЛЕТ 14

1. Деревья

2. Написать программу для умножения двух прямоугольных матриц
3. Написать программу для нахождения длины самой длинной возрастающей цепочки элементов в последовательности чисел (без массивов)

БИЛЕТ 15

1. Файловый ввод-вывод
2. Написать программу для проверки упорядоченности последовательности целых чисел (без массива)
3. Написать программу, которая в квадратной матрице вычислит сумму элементов, которые находятся одновременно и под главной и под побочной диагоналями

БИЛЕТ 16

1. Прямой доступ к файлам
2. Написать программу для проверки того, что элементы массива образуют арифметическую прогрессию
3. Написать программу для поворота квадратной матрицы на 180 градусов

БИЛЕТ 17

1. Двумерные массивы
2. Написать программу, которая в заданном массиве сдвинет все положительные элементы к началу, не меняя их взаимного расположения.
3. Написать программу, которая в двух символьных строках найдет и напечатает самую длинную одинаковую цепочку подряд идущих символов

БИЛЕТ 18

1. Данные, их типы и описание
2. Написать программу для умножения двух заданных многочленов
3. Написать функцию для перевода целого числа в двоичную систему

БИЛЕТ 19

1. Основные типы операторов
2. Дана символьная строка. Написать программу для печати каждого слова буквами наоборот
3. Написать программу для нахождения пересечения упорядоченных массивов целых чисел.

Вопросы для текущего контроля

1. Алфавит. Кодовая страница. ASCII-код. Комментарии. Идентификаторы. Ключевые слова.
2. Определение типа. Операции и операнды. Предопределенные и пользовательские типы. Переменные и константы. Оператор-определение.
3. Объекты типа int. Представление объектов типа int. Арифметические операции. Операции сравнения. Логические операции. Операции присваивания. Тернарная операция. Побитовые операции. Типы short и unsigned int.
4. Объекты типа float. Объекты типа double и long double. Представление объектов с плавающей точкой. Операции над объектами с плавающей точкой.
5. Определение типа char. Представление объектов типа char. Операции с объектами типа char.
6. Неявные и явные преобразования типов. Ранги типов. Использование префиксов в идентификаторах.
7. Определение выражения. Порядок выполнения операций. Приоритеты операций и законы ассоциативности.
8. Оператор-определение
9. Оператор-выражение
10. Составной оператор
11. Условные операторы
12. Операторы цикла

13. Пустой оператор
14. Безусловные операторы
15. Оператор многоальтернативного выбора
16. Обязательные компоненты определения функции. Использование оператора return; Примеры функций. Формула Герона. Стандартные функции.
17. Вызов функции. Фактические параметры. Прототип функции. Определение программы. Заголовочные файлы. Пример табулирования функции.
18. Операция получения адреса. Операция разыменования.
19. Адрес объекта в качестве параметра функции. Функция Change().
20. Стандартные функции printf() и scanf()
21. Строковые константы. Доступ к строковой константе и завершающий нуль-символ. Прототип функции printf(). Команды форматирования функции printf(). Прототип функции scanf(). Команды форматирования функции scanf().
22. Определение и виды областей видимости. Локальные и глобальные идентификаторы. Примеры. Скрытие глобальных идентификаторов. Операция ::. Время жизни объектов. Автоматические и статические идентификаторы.
23. Определение массива. Примеры. Выделение памяти.
24. Указатель на константу. Указатель-константа. Адрес массива. Адреса элементов массива. Операции с указателями. Операция индексирования.
25. Передача массива в функцию. Вычисление среднего арифметического. Сортировка выбором минимального элемента. Вычисление суммы двух векторов.
26. Статическое управление памятью. Отличие динамической памяти от статической. Операции new и delete. Два варианта функции сложения векторов.
27. Объединение массивов с одинаковой длиной. Объединение массивов с разной длиной.
28. Определение таблицы. Массив из элементов таблицы. Доступ к элементам таблицы. Использование стандартной функции rand. Массив из строк таблицы. Выделение и освобождение памяти.
29. Строковые константы. Указатель на строковую константу и указатель-константа. Символьные строки и их применение. Стандартные функции gets() и puts(), strlen(), strcmp(), strcpy(), strcat(), getch() и putchar().
30. Рекурсия
31. Указатели на функции
32. Перегрузка функций
33. Обобщенные функции
34. Определение структурного типа
35. Свойства структурных типов
36. Перечисления
37. Диск. Поверхность, дорожка, сектор. Файл и его имя. Каталоги. Полное имя файла. Вывод и ввод. Время поиска и время ожидания. Скорость вращения.
38. Текстовые файлы. Графические файлы. Файлы с приложениям. Форматы файлов.
39. Буферизация ввода и вывода. Структура типа FILE. Открытие и закрытие файлов. Бинарные и текстовые файлы. Текущая позиция.
40. Запись и чтение объектов абстрактных типов
41. Запись и чтение строк
42. Чтение и запись объектов типа char

### 7.1. Основная литература:

1. Андрианова, А.А. Объектно-ориентированное программирование на C++. Часть 1. / А.А.Андрианова, Л.Н.Исмагилов, Т.М.Мухтарова. - Казань: Казанский федеральный университет, 2010.- 124 с.
2. Андрианова, А.А. Объектно-ориентированное программирование на C++. Часть 2. / А.А.Андрианова, Л.Н.Исмагилов, Т.М.Мухтарова. - Казань: Казанский федеральный университет, 2010.- 116 с.
3. Липачёв Е.К. Технология программирования. Базовые конструкции C/C++.  
Учебно-справочное пособие // Казань: Казан. ун-т, 2012. " 142 с  
[http://kpfu.ru/publication?p\\_id=47437](http://kpfu.ru/publication?p_id=47437)
- 4.Зиборов В. В. Visual C# 2010 на примерах. ? СПб.: БХВ-Петербург, 2011. ? 423 с. - ISBN 978-5-9775-0698-4.  
<http://www.znanium.com/bookread.php?book=355304>
- 5.Монахов, В. В. Язык программирования Java и среда NetBeans / В. Монахов. ? 2-е изд., перераб. и доп. ? СПб.: БХВ-Петербург, 2009. ? 718 с.. - ISBN 978-5-9775-0424-9.  
<http://www.znanium.com/bookread.php?book=350724>
- 6.Бенкен Е. С. PHP, MySQL, XML: программирование для Интернета / Е. С. Бенкен. ? 3-е изд., перераб. и доп. ? СПб.: БХВ-Петербург, 2011. ? 304 с.: ил. + (CD-ROM) ISBN 978-5-9775-0724-0  
<http://www.znanium.com/bookread.php?book=352144>
- 7.Культин Н. Б. Основы программирования в Microsoft Visual C# 2010. ? СПб.: БХВ-Петербург, 2011. ? 364 с. ? Режим доступа: <http://www.znanium.com/bookread.php?book=351294>

### 7.2. Дополнительная литература:

1. Java : объектно-ориентированное программирование : для магистров и бакалавров : базовый курс по объектно-ориентированному программированию : [учебное пособие] / А. Н. Васильев .? Санкт-Петербург [и др.] : Питер, 2013 .? 395, [1] с. : ил. ; 24 .? (Учебное пособие) (Стандарт третьего поколения) .? По номеру записи 1878254 .? Библиогр.: с. 377 (11назв.) .? Алф. указ. в конце кн.
2. Программирование на языке C# : учебно-методическое пособие / И. Л. Александрова, Д.Н. Тумаков ; Казан. федер. ун-т, Ин-т вычисл. мат. и информ. технологий .? Казань : [б. и.], 2011 .? 103 с. ; 21.
3. C++: Объектно-ориентированное программирование : практикум:учеб.пособие для студ.вузов / Т. А. Павловская, Ю. А. Щупак .? СПб. : Питер, 2006 .? 265 с. : ил. ? Алф.указ. ? ISBN 5-94723-842-x : p.156.40.

### **7.3. Интернет-ресурсы:**

Интернет-портал образовательных ресурсов КФУ - <http://www.kfu-elearning.ru/>

Интернет-портал образовательных ресурсов по ИТ - <http://www.intuit.ru>

Интернет-портал по программным средствам Microsoft - <http://msdn.microsoft.com/ru-ru/vcsharp>

Интернет-портал со статьями по алгоритмике и программированию - <http://algotlist.manual.ru/>

Электронная библиотека по техническим наукам - <http://techlibrary.ru>

### **8. Материально-техническое обеспечение дисциплины(модуля)**

Освоение дисциплины "Программирование" предполагает использование следующего материально-технического обеспечения:

Компьютерные классы (9 классов) лаборатории малой вычислительной техники Института ВМ и ИТ, оснащенные мультимедийным оборудованием.

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 080500.62 "Бизнес-информатика" .

Автор(ы):

Фофанов В.Б. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.

Рецензент(ы):

Тагиров Р.Р. \_\_\_\_\_

Миссаров М.Д. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.