**Программа учебной дисциплины «Программирование»**

*Утверждена*
*Академическим руководителем*

*_____ Н.В. Асеева*

*_____ 20_____*

| | |
|---|---|
| Автор | Дёмкин В.М., к.т.н., доцент |
| Число кредитов | 4 |
| Контактная работа (час.) | 40 |
| Самостоятельная работа (час.) | 112 |
| Курс | 2 курс |
| Формат изучения дисциплины | без использования онлайн курса |

## I. ЦЕЛЬ, РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ И ПРЕРЕКВИЗИТЫ

Целями освоения дисциплины «Программирование» являются овладение основами алгоритмизации и императивного программирования на языке C++, приобретение навыков разработки программного кода с использованием современных инструментальных средств для платформ MS Windows и Linux.

В результате освоения дисциплины студент должен:

**знать:**

- методологию анализа и проектирования классических типов алгоритмов (линейных, условных, циклических, рекурсивных и эвристических). Знать теоретические и практические аспекты перехода от содержательной постановки задачи к разработке алгоритма её решения. Чётко представлять себе алгоритмическую систему, в рамках которой составляется алгоритм, представлять и учитывать возможности исполнителя алгоритма (человека или компьютера);

**уметь:**

- применять полученные знания к решению вопросов составления и документирования алгоритмов, разработки программ и их сопровождения, использования современных инструментальных средств и технологий;

**владеть:**

- методологией разработки программ на основе императивного подхода. Приобрести навыки разработки программного кода с использованием современных инструментальных средств для платформ MS Windows и Linux.

Для освоения учебной дисциплины студенты должны владеть следующими знаниями и компетенциями:

- навыками работы с компьютером как средством обработки информации;
- современными методами и средствами информационных технологий;
- культурой мышления, навыками анализа и моделирования;
- основами информатики, навыками алгоритмизации и программирования выпускника средней школы;

– навыками работы с системами программирования на этапах разработки и отладки программного кода.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин:

– «Объектно-ориентированное программирование»;
– «Численные методы моделирования»;
– «Базы и хранилища данных»;
– «Управление данными»;
– «Моделирование процессов и систем»;
– «Функциональное программирование и интеллектуальные системы»;
– «Распределённые информационные системы»;
– «Имитационное моделирование».

## II. СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### Тема 1. АЛГОРИТМЫ И АЛГОРИТМИЗАЦИЯ

Понятие алгоритма. Алгоритмические системы. Исполнители алгоритма. Языки для формулировки алгоритмов: естественный язык, язык блок-схем и языки символического кодирования. Понятие о математической модели. Этапы построения математической модели. Алгоритмизация и этапы разработки алгоритма. Типы алгоритмов: линейный, условный, циклический, рекурсивный и эвристический.

### Тема 2. СИСТЕМЫ ПРОГРАММИРОВАНИЯ И C++

Технологии разработки программного кода на C++. Платформа MS Windows и C++. Интегрированная среда разработки программного обеспечения MS Visual Studio. Платформа Linux и C++. Интерфейс командной строки. Командный интерпретатор bash. Текстовый редактор gedit. Компилятор gcc.

### Тема 3. ПАРАДИГМЫ ПРОГРАММИРОВАНИЯ И C++

C++ и поддержка основных парадигм программирования: императивной, модульной, объектно-ориентированной и обобщённой. Обзор языков программирования C и C++. Структура программы на C и C++. Заголовочные файлы и компоновка программы. Стандартные библиотеки C и C++. Ввод и вывод в C и C++. Математические вычисления в C и C++.

### Тема 4. ЛЕКСИЧЕСКИЕ ОСНОВЫ C++

Алфавит и лексемы. Идентификаторы и ключевые слова. Пространство имён. Операторы и операции. Разделители.

### Тема 5. ТИПЫ И ОБЪЯВЛЕНИЯ

Фундаментальные типы: логический тип, символьные типы, целые типы, типы с плавающей точкой. Тип void. Типы, определяемые пользователем. Указатели. Массивы. Ссылки. Объявления и определения. Имена. Константы. Область видимости. Инициализация. Указатели и адресная арифметика. Классификация указателей. Указатели на объекты. Операции над указателями. Указатели и константы.

### Тема 6. ОПЕРАТОРЫ И ИНСТРУКЦИИ C++

Обзор стандартных операторов. Обзор инструкций. Инструкция-объявление. Инструкция-выражение. Составная инструкция. Помеченная инструкция. Инструкция выбора. Инструкция итерации. Инструкция передачи управления. Блок try.

### Тема 7. ФУНКЦИИ

Объявления и определения функций. Аргументы функций. Типы функций. Вызовы функций. Аргументы по умолчанию. Неуказанное количество аргументов. Рекурсивные функции. Встраивание функций. Перегрузка функций. Указатели на функции.

**Тема 8. МАССИВЫ**

Объявления массивов. Массивы и указатели. Одномерные и многомерные массивы. Массивы как аргументы функции.

**Тема 9. ФАЙЛЫ**

Файловые потоки. Текстовые и двоичные файлы. Ввод-вывод текстовых файлов.

## III. ОЦЕНИВАНИЕ

**Критерии оценки знаний, навыков**

Оценки по всем формам контроля выставляются по 10-балльной шкале. Требования к ответу студента соотносятся с указанными в пункте 3 компетенциями.

**Контрольная работа и домашнее задание:**

высшая оценка в 9 баллов (10 баллов только в исключительных случаях) проставляется при полностью правильных ответах на вопросы и отличном выполнении заданий (правильном решении задачи, чётком и исчерпывающем её представлении);

почти отличная оценка в 8 баллов проставляется при полностью правильных ответах на вопросы и отличном выполнении заданий, но при отсутствии чёткого и исчерпывающего представления решаемой задачи;

оценка в 7 баллов проставляется при правильных ответах на вопросы и правильном решении задачи, но при наличии отдельных неточностей в ответах на вопросы;

оценка в 6 баллов проставляется при наличии отдельных неточностей в ответах на вопросы (включая грамматические ошибки) или неточностях в решении задачи непринципиального характера (описки и случайные ошибки);

оценка в 5 баллов проставляется в случаях, когда в ответах на вопросы и в решении задачи имеются неточности и ошибки, свидетельствующие о недостаточном понимании изучаемой дисциплины и требующие дополнительного обращения к учебным материалам;

оценка в 4 балла проставляется при наличии серьёзных ошибок в ответах на вопросы и в решении задачи, что свидетельствует о наличии пробелов в знании изучаемой дисциплины;

оценка в 3 балла проставляется при наличии лишь отдельных положительных моментов в ответах на вопросы и в решении задач, говорящих лишь о потенциальной возможности в последующем более успешного выполнения заданий; оценка в 3 балла, как правило, ведёт к повторному решению дополнительной задачи;

оценка в 2 балла проставляется при полном отсутствии положительных моментов в ответах на вопросы и в решении задачи и, как правило, ведёт к повторному написанию контрольной работы в целом;

оценка в 1 балл проставляется в тех случаях, когда наряду с неправильными ответами на вопросы и решением задачи имеют место какие-либо демонстративные проявления безграмотности или неэтичное отношение к изучаемой дисциплине.

**Экзамен:**

На экзамене, представляющем собой письменные ответы на вопросы и решение задачи с последующим собеседованием, оценка проставляется следующим образом:

высшая оценка в 9 баллов (10 баллов только в исключительных случаях) проставляется при отличном выполнении заданий (полных, с примерами и возможными обобщениями ответах на вопросы, при правильном решении задачи и детальном её представлении);

почти отличная оценка в 8 баллов проставляется при полностью правильных ответах на вопросы и решении задачи, но при отсутствии примеров и обобщений, а также детального представления решаемой задачи;

оценка в 7 баллов проставляется при правильных ответах на вопросы и правильном решении задачи, но при отсутствии пояснений и обобщений, а также детального представления решаемой задачи;

оценка в 6 баллов проставляется при наличии отдельных неточностей в ответах на вопросы или неточностях в решении задачи непринципиального характера (описки и случайные ошибки);

оценка в 4-5 баллов проставляется в случаях, когда в ответах на вопросы и в решении задачи имеются существенные неточности и ошибки, свидетельствующие о недостаточном понимании изучаемой дисциплины;

оценка в 2-3 балла проставляется при наличии лишь отдельных положительных моментов в ответах на вопросы и в решении задачи;

оценка в 1 балл проставляется в тех случаях, когда наряду с неправильными ответами на вопросы и решением задачи имеют место какие-либо демонстративные проявления безграмотности или неэтичное отношение к изучаемой дисциплине.

**Порядок формирования оценок по дисциплине**

Преподаватель оценивает работу студентов на практических занятиях и самостоятельную работу, выставляя баллы за активность в аудитории, контрольные работы и домашние задания. Оценки за все виды работ выставляются в рабочей электронной ведомости. Критерии их оценивания приведены в пункте 6.1. Доступ к рабочей электронной ведомости обеспечивается благодаря ресурсам компьютерного класса и сети Интернет.

Результирующая оценка $О_{аудиторная}$ за работу в аудитории определяется перед промежуточным или итоговым контролем.

Результирующая оценка $О_{текущий}$ за текущий контроль вычисляется по следующей формуле, где $О_{к/р}$ и $О_{д/з}$ – соответственно оценки за контрольную работу и домашнее задание:

$$О_{текущий} = 0,8 \cdot О_{к/р} + 0,2 \cdot О_{д/з}$$

Результирующая оценка $О_{итоговый}$ за итоговый контроль в форме экзамена вычисляется по следующей формуле, где $О_{экзамен}$ – оценка за работу непосредственно во время экзамена:

$$О_{итоговый} = 0,5 \cdot О_{экзамен} + 0,4 \cdot О_{текущий} + 0,1 \cdot О_{аудиторная}$$

В диплом выставляется результирующая оценка $О_{дисциплина}$, которая вычисляется по следующей формуле:

$$О_{дисциплина} = 0,2 \cdot О_{итоговый_1} + 0,2 \cdot О_{итоговый_2} + 0,3 \cdot О_{итоговый_3} + 0,3 \cdot О_{итоговый_4}$$

## IV. ПРИМЕРЫ ОЦЕНОЧНЫХ СРЕДСТВ

**Оценочные средства для текущего контроля студента**

**Перечень вопросов к промежуточной форме контроля (зачёт)**

1. Понятие алгоритма и алгоритмического процесса.
2. Алгоритмические системы.
3. Исполнители алгоритма.
4. Языки для формулировки алгоритмов.
5. Естественный язык.
6. Язык блок-схем.
7. Язык символического кодирования.
8. Понятие о математической модели.
9. Этапы построения математической модели.
10. Алгоритмизация и этапы разработки алгоритма.
11. Типы алгоритмов.
12. Линейный алгоритм.
13. Условный алгоритм.
14. Условный алгоритм с двумя ветвями.
15. Условный алгоритм с множеством ветвей.
16. Циклический алгоритм.
17. Цикл с предусловием.
18. Цикл с постусловием.
19. Рекурсивный алгоритм.
20. Эвристический алгоритм.

**Оценочные средства для промежуточной аттестации**

### Перечень вопросов к итоговой форме контроля (экзамен)

1. C++ и поддержка основных парадигм программирования: императивной, модульной, объектно-ориентированной и обобщённой.
2. Структура программы на C и C++.
3. Заголовочные файлы и компоновка программы.
4. Стандартные библиотеки C и C++.
5. Ввод и вывод в C и C++.
6. Математические вычисления в C и C++.
7. Алфавит и лексемы C++.
8. Пространство имён.
9. Операторы и операции.
10. Фундаментальные типы: логический тип, символьные типы, целые типы, типы с плавающей точкой.
11. Тип void.
12. Типы, определяемые пользователем.
13. Указатели.
14. Массивы.
15. Ссылки.
16. Объявления и определения.
17. Имена.
18. Константы.
19. Область видимости.
20. Инициализация.
21. Указатели и адресная арифметика.
22. Операции над указателями.
23. Стандартные операторы C++.
24. Инструкции C++: инструкция-объявление, инструкция-выражение, составная инструкция, помеченная инструкция, инструкция выбора, инструкция итерации, инструкция передачи управления, блок try.
25. Объявления и определения функций.
26. Аргументы функций.
27. Типы функций.
28. Вызовы функций.

29. Аргументы по умолчанию.
30. Неуказанное количество аргументов.
31. Встраивание функций.
32. Перегрузка функций.
33. Указатели на функции.
34. Объявления массивов.
35. Массивы и указатели.
36. Одномерные и многомерные массивы.
37. Массивы как аргументы функции.
38. Файловые потоки.
39. Текстовые и двоичные файлы.
40. Ввод-вывод текстовых файлов.

## V. РЕСУРСЫ

### 5.1 Основная литература

1. Голицына, О.Л. Языки программирования [Электронный ресурс]: учебное пособие / О.Л.Голицына, Т.Л.Партыка, И.И.Попов; ЭБС Знаниум. - 2-е изд., перераб. и доп. - М.: Форум, 2010. - 400 с.: ил. - (Профессиональное образование). - ISBN 978-5-91134-442-9. - Режим доступа: http://znanium.com/bookread.php?book=226043. - Загл. с экрана

### 5.2 Дополнительная литература

1. Канцедал, С.А. Алгоритмизация и программирование [Электронный ресурс]: учебное пособие / С.А.Канцедал; ЭБС ЗНАНИУМ. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 352 с.: ил. - (Профессиональное образование). - ISBN 978-5-8199-0355-1. - Режим доступа: http://znanium.com/bookread.php?book=429576. - Загл. с экрана. Гриф МО РФ
2. Немцова, Т.И. Программирование на языке высокого уровня. Программирование на языке С++ [Электронный ресурс]: учебное пособие / Т.И.Немцова [и др.]; под ред. Л.Г.Гагариной; ЭБС ЗНАНИУМ. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с.: ил. - (Профессиональное образование). - ISBN 978-5-8199-0492-3. - Режим доступа: http://znanium.com/bookread.php?book=244875. - Загл. с экрана.
3. Дёмкин, В.М. Основы алгоритмизации и императивного программирования: учебное пособие / В.М.Дёмкин. - Н.Новгород: НФ ГУ-ВШЭ, 2003. - 144 с.

### 5.3 Программное обеспечение

| № п/п | Наименование | Условия доступа |
|---|---|---|
| 1. | Microsoft Visual Studio 2013 Prof | *Из внутренней сети университета (договор)* |
| 2. | Microsoft Office 2013 Prof + | *Из внутренней сети университета (договор)* |

### 5.4 Профессиональные базы данных, информационные справочные системы, интернет-ресурсы (электронные образовательные ресурсы)

| № п/п | Наименование | Условия доступа |
|---|---|---|
| | ***Профессиональные базы данных, информационно-справочные системы*** | |
| 1. | Электронно-библиотечная система Юрайт | URL: https://biblio-online.ru/ |
| | ***Интернет-ресурсы (электронные образовательные ресурсы)*** | |

| | | |
|---|---|---|
| 1. | Открытое образование | URL: https://openedu.ru/ |

### 5.5 Материально-техническое обеспечение дисциплины

Компьютерный класс, позволяющий обеспечить проведение практических занятий по программированию для платформ MS Windows и Linux.