

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Национальный исследовательский ядерный университет «МИФИ»  
**Димитровградский инженерно-технологический институт –**  
филиал федерального государственного автономного образовательного учреждения высшего  
образования «Национальный исследовательский ядерный университет «МИФИ»  
**(ДИТИ НИЯУ МИФИ)**

**УТВЕРЖДАЮ:**

Заместитель руководителя

\_\_\_\_\_ Т.И. Романовская

« \_\_\_\_ » \_\_\_\_\_ 2021г

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**Программирование**

Направление подготовки \_\_\_\_\_ 09.03.01 Информатика и вычислительная техника

Квалификация выпускника \_\_\_\_\_ бакалавр

Профиль \_\_\_\_\_ *Программное обеспечение средств вычислительной техники и автоматизированных систем*

Форма обучения \_\_\_\_\_ очная

Выпускающая кафедра \_\_\_\_\_ Информационных технологий

Кафедра-разработчик рабочей программы \_\_\_\_\_ Информационных технологий

Семестр	Трудоемкость час. (ЗЕТ)	Лекций, час.	Практич. занятий, час.	Лаборат. работ, час.	СРС, час.	Форма промежуточного контроля (экз., час./зачет)
2	216(6)	34	34	34	78	Экз
3	144(4)	17	17	17	57	Экз.
<b>Итого</b>	<b>360(10)</b>	<b>51</b>	<b>51</b>	<b>51</b>	<b>135</b>	

Димитровград  
2021 г.

## СОДЕРЖАНИЕ

1 ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	3
2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО.....	3
3 ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	4
4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	5
5 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ.....	11
6 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ВХОДНОГО И ТЕКУЩЕГО КОНТРОЛЯ, ПРОМЕЖУТОЧНОЙ И ИТОГОВОЙ АТТЕСТАЦИИ (АННОТАЦИЯ).....	11
7 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	12
8 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	12
9 ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ.....	12

## 1 ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Целью** дисциплины является необходимость формирования у студентов представлений о современном состоянии языков программирования, о современном программном обеспечении, операционных системах и средствах для разработки программ различного уровня сложности.

**Задачей** дисциплины является развитие практических навыков по разработке программ с использованием любых языков программирования и сред для разработки программ.

Дисциплина обеспечивает совершенствование навыков, полученных при изучении основ информатики. Акцент смещается на стадию разработки алгоритма, под которую подводится теоретическая база. В основу закладывается понятие эффективности алгоритма. Активно используются сведения, полученные студентами из курсов высшей и дискретной математики, информатики.

## 2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина Программирование относится к базовой части блока 1 профессионального модуля учебного плана.

Необходимыми условиями для освоения дисциплины являются:

*знание* базовых понятий информатики и вычислительной техники, роли и значения информатики в современном обществе, форм представления и преобразования информации в компьютере;

*умения* применять вычислительную технику для решения практических задач, оперировать элементами алгебры логики;

*владение* навыками работы на персональном компьютере.

Таблица 2.1 - Перечень предшествующих и последующих дисциплин, формирующих общекультурные и профессиональные компетенции

Код	Наименование компетенции	Предшествующие дисциплины	Последующие дисциплины (группы дисциплин)
Профессиональные компетенции			
ПК-3	способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования	Программирование на Delphi Промышленная робототехника Математическое программное обеспечение Основы проектной деятельности	Функциональное и логическое программирование Объектно-ориентированное программирование Технология разработки программного обеспечения Web-программирование на ASP.NET Математическая логика и теория алгоритмов Мультимедийные технологии Структуры и алгоритмы обработки данных Программирование под платформу .Net Теория языков программирования и методы трансляции Основы моделирования систем Производственная практика (преддипломная) Итоговая государственная аттестация
ПК-4	Способность осуществлять	Программирование на	Операционные системы

	организацию и управление проектами в области информатики и вычислительной техники в соответствии с действующими правовыми нормами и требованиями заказчика	Delphi	Сети и телекоммуникации Базы данных Производственная практика (преддипломная)
<b>Дополнительные компетенции</b>			
ДК-3	способность решать задачи производственной и технологической деятельности на профессиональном уровне, включая разработку алгоритмических решений в области прикладного программирования;	Теория вероятностей и математическая статистика Математическое программное обеспечение	Компьютерное моделирование Программирование на языке Си Объектно-ориентированное программирование Технология разработки программного обеспечения
ДК-7	использовать ЭВМ для управления и обработки информации	информатика Программирование на Delphi	Защита информации Основы моделирования систем
<b>Общепрофессиональные компетенции</b>			
ОПК-9	способность осваивать методики использования программных средств для решения практических задач;	Программирование на Delphi Математическое программное обеспечение	Базы данных Программирование Web-технологии Функциональное и логическое программирование Объектно-ориентированное программирование Web-программирование на ASP.NET Программирование под платформу .Net Компьютерное моделирование Теория языков программирования и методы трансляции Инструментальные средства информационных систем Технологии программирования в сетях Производственная (технологическая) Производственная практика (преддипломная) Итоговая государственная аттестация

### 3 ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на формирование элементов компетенций в соответствии с ОС НИЯУ МИФИ и ОП ВО по данному направлению подготовки (специальности).

Таблица 3.1 - Перечень планируемых результатов обучения по дисциплине

Планируемые результаты освоения ОП (компетенции), достижение которых обеспечивает дисциплина	Перечень планируемых результатов обучения по дисциплине
--	---

ПК-3	способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования	<i>Знать:</i> современные технические и программные средства взаимодействия с ЭВМ; <i>Уметь:</i> ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, <i>Владеть:</i> языками процедурного и объектно-ориентированного программирования, навыками разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков программирования высокого уровня
ПК-4	Способность осуществлять организацию и управление проектами в области информатики и вычислительной техники в соответствии с действующими правовыми нормами и требованиями заказчика	<i>Знать:</i> современные технические и программно-методические комплексы; <i>Уметь:</i> разрабатывать основные документы, работать с современными системами Программирования <i>Владеть:</i> опытом использование современных технологий программирования, тестирования и документирования программ.
ДК-3	способность решать задачи производственной и технологической деятельности на профессиональном уровне, включая разработку алгоритмических решений в области прикладного программирования;	<i>Знать:</i> технологию разработки алгоритмов и программ; <i>Уметь:</i> решать задачи производственной и технологической деятельности на профессиональном уровне; <i>Владеть:</i> умением использовать современные инструментальные средства и технологии программирования.
ДК-7	использовать ЭВМ для управления и обработки информации	<i>Знать:</i> методы отладки и решения задач на ЭВМ в различных режимах; <i>Уметь:</i> ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, <i>Владеть:</i> умением разрабатывать интерфейсы «человек-электронно-вычислительная машина».
ОПК-9	способность осваивать методики использования программных средств для решения практических задач;	<i>Знать:</i> методы структурного и модульного программирования; <i>Уметь:</i> описывать основные структуры данных, реализовывать методы обработки данных; <i>Владеть:</i> опытом работы в различных средах программирования.

## 4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1 Структура дисциплины

Общая трудоемкость (объем) дисциплины составляет 10 зачетных единиц (ЗЕТ),

Таблица 4.1

## Объём дисциплины по видам учебных занятий

Вид учебной работы	Всего, зачетных единиц (акад. часов)	Семестр	
		2	3
<b>Общая трудоемкость дисциплины</b>	360	216	144
<b>Контактная работа с преподавателем:</b>			
занятия лекционного типа	54	34	18
в том числе: практические занятия	54	34	18
лабораторные работы	54	34	18
<b>Самостоятельная работа обучающихся:</b>	141	<b>87</b>	<b>54</b>
изучение теоретического курса	100	66	34
задачи	41	21	20
<b>Вид промежуточной аттестации (экзамен)</b>	экзамен	экзамен	экзамен

## Распределение учебной нагрузки по разделам дисциплины

Таблица 4.2

№ раздела	Наименование раздела дисциплины	Виды учебной нагрузки и их трудоемкость, акад. часы					Формируемые компетенции
		Лекции	Практические занятия	Лабораторные работы	Самостоятельная работа	Всего часов	
1 семестр							
1.	<b>Тема 1.</b> Введение в программирование и языки. Обзор современного состояния компьютерной техники.	2	2	-	10	4	ПК-3 , ПК-4,
2.	<b>Тема 2.</b> Основные этапы решения задач на ЭВМ. Алгоритмы. Способы записи алгоритма.	6	6	6	13	29	ПК-3 , ПК-4, ДК-3 , ДК-7 ОПК-9
3.	<b>Тема 3.</b> Обзор современных языков программирования. Данные и программы. Критерии качества программы.	6	6	6	15	28	ПК-3 , ПК-4, ДК-3 , ДК-7
4.	<b>Тема 4.</b> Структура программы на языке высокого уровня. Стандартные типы данных.	6	6	5	15	27	ДК-3 , ДК-7, ОПК-9
5.	<b>Тема 5.</b> Операторы языка: присваивания, ветвление, повторение, итерация.	6	6	8	18	31	ПК-3 , ПК-4, ДК-3 , ДК-7 ОПК-9
6.	<b>Тема 6.</b> Типы данных, определяемые пользователем: массивы, структуры, объединения, файлы.	8	8	9	19	45	ПК-3 , ПК-4, ДК-3 , ДК-7 ОПК-9
	<b>Итого</b>	<b>34</b>	<b>34</b>	<b>34</b>	<b>87</b>		

2 семестр							
7.	<b>Тема 7.</b> Способы конструирования программ. Процедуры и функции	2	2	2	7	20	,ПК-3 , ПК-4, ДК-3 , ДК-7
8.	<b>Тема 8.</b> Разработка простейшего интерфейса. Диалоговые программы.	3	3	3	7	21	,ПК-3 , ПК-4, ДК-3 , ДК-7
9.	<b>Тема 9.</b> Динамические структуры данных. Списки: основные виды и способы реализации.	4	4	4	7	26	ОПК-9 , ДК-3 , ДК-7
10.	<b>Тема 10.</b> Сортировка массивов и списков.	3	3	3	9	21	ДК-3 , ДК-7
11.	<b>Тема 11.</b> Программирование рекурсивных алгоритмов.	3	3	3	12	34	ПК-3 , ПК-4,ОПК-9,
12.	<b>Тема 12.</b> Способы конструирования и верификации программ. Модульные программы. Основы доказательства правильности.	3	3	3	12	30	ПК-3 , ПК-4, ДК-3 , ОПК-2,
	<b>Итого</b>	<b>1818</b>	<b>18</b>	<b>18</b>	<b>54</b>		

#### 4.2 Содержание дисциплины

Удельный вес проводимых в активных и интерактивных формах проведения аудиторных занятий по дисциплине составляет \_\_\_\_\_ %.

#### Лекционный курс

Таблица 4.3

№ лекции	Номер раздела	Тема лекции и перечень дидактических единиц	Трудоемкость, акад. часов	
			всего	в том числе с использованием интерактивных образовательных технологий
2 семестр				
1.	1.	<b>Тема 1.</b> Введение в программирование и языки. Обзор современного состояния компьютерной техники. 1.1. Введение в программирование и языки. Обзор современного состояния ПК и характеристик ЭВМ. Характеристики ЭВМ и её компонент. Единицы информации. Операционные системы, основные характеристики. Дисковые операционные системы (DOS), многооконные операционные системы (Windows).Понятие файла в операционной системе. Команды в операционных системах.	2	
2-4	2	<b>Тема 2.</b> Основные этапы решения задач на ЭВМ. Алгоритмы. Способы записи алгоритма. 2.1. Алгоритмы и программы. Основные правила разработки алгоритмов для решения задач. Базовые алгоритмические структуры: следование, развилка, повторение. Способы их изображения. Примеры алгоритмов.	6	
5-7	3.	<b>Тема 3.</b> Обзор современных языков программирования. Данные и программы. Критерии качества программы.	6	

		3.1. Сравнительная характеристика языков программирования. Синтаксис, семантика. Общая характеристика языков программирования. Сравнение развития языков в представлении данных и способах реализации алгоритмов. Представление основных управляющих структур программирования.		
8-10	4	<b>Тема 4.</b> Структура программы на языке высокого уровня. Стандартные типы данных. 4.1. Основные этапы разработки программ. Структура языковой программы. Средства для написания программ: алфавит, лексика, синтаксис языка. Термины языка. Система типов данных. Стандартные и скалярные типы данных. Диапазоны изменения различных типов данных. Символьный тип, ограниченные и перечисляемые типы. Основные операции над простыми типами данных. 4.2. Константы в языке. Простые, типизированные, адресные, структурированные константы. Определение переменных. Преобразование типов данных. Примеры использования рассмотренных структур. Выражение в языке для разных типов данных.	6	
11-13	5.	<b>Тема 5.</b> Операторы языка : присваивания, ветвление, повторение, итерация. 5.1. Операторы языка: присваивания, ввод, вывод. Простейшие программы. Организация повторений работы программы. Директивы компилятора для обработки программ. Составной и условный операторы. 5.2. Оператор выбора. Работа со строками. Внутреннее представление строк. Простейшие циклы. Циклы с итерациями и циклы в циклах. Примеры использования языковых конструкций.	6	
14-17	6.	<b>Тема 6.</b> Типы данных, определяемые пользователем: массивы, структуры, объединения, файлы. 6.1. Структуры данных и алгоритмы. Массив как структура данных. Сортировка массивов. 6.2. Структуры, объединения. 6.3. Организация работы с файлами. Типы файлов. Примеры.	8	
		<b>Итого</b>	<b>34</b>	
3 семестр				
1-2	7.	<b>Тема 7.</b> Способы конструирования программ. Процедуры и функции. 7.1. Процедуры и функции в языке. Основные понятия. Принципы использования процедур и функций в программах. Параметры процедур и функций.	2	
3-4	8.	<b>Тема 8.</b> Разработка простейшего интерфейса. Диалоговые программы. 8.1. Проектирование программ. Постановка задачи и спецификация программы. Принципы структурного программирования. Этапы разработки программ. Организация интерфейса.	2	



5-7	9.	<b>Тема 9.</b> Динамические структуры данных. Списки: основные виды и способы. 9.1. Динамические структуры данных. Указатели и ссылки. Динамические массивы. Построение списков разных типов: стеки, очереди.	3	
8-11	10.	<b>Тема 10.</b> Сортировка массивов и списков.	8	
12-15	11.	<b>Тема 11.</b> Программирование рекурсивных алгоритмов. 11.1. Понятие рекурсии. Использование рекурсии для записи решений. Древовидные структуры. Бинарные деревья.	8	
16-18	12.	<b>Тема 12.</b> Способы конструирования программ. Модульные программы. Основы доказательства правильности. 12.1 Модули в языке: назначение, структура, трансляция, тестирование. Особенности использования модулей. Стандартные модули. 12.2. Понятие дружественного интерфейса. Создание элементов меню. Использование шаблонов меню. Оценка качества разработанных программ. Основы доказательства правильности программ.	6	
<b>Итого:</b>			<b>18</b>	

### Практические занятия

Таблица 4.4

№ занятия	Номер раздела	Наименование практического занятия и перечень дидактических единиц	Трудоемкость, акад. часов	
			всего	в том числе с использованием интерактивных образовательных технологий
<b>1 семестр</b>				
1	1	Разработка алгоритмов.	4	
2-3	2	Разработка простейших программ.	6	
3	3	Разработка программ с разветвлением.	2	
4	4	Разработка программ с циклами.	6	
4-5	5	Работа со строками.	4	
6-8	6	Работа с массивами.	12	
<b>Итого:</b>			<b>34</b>	
<b>2 семестр</b>				
1	7	Работа со структурами.	2	
2-3	8	Работа с файлами.	3	
4-5	9	Сортировка массивов.	4	
5-6	10	Принципы использования процедур и функций в программах.	3	
6-7	11	Использование рекурсии для записи решений.	3	

7-8	12	Динамические структуры данных.	3	
8-9	12	Модули.	3	
		Итого	<b>18</b>	

### Лабораторные работы

Таблица 4.5

№ занятия	Номер раздела	Наименование лабораторной работы и перечень дидактических единиц	Трудоемкость, акад. часов	
			всего	в том числе с использованием интерактивных образовательных технологий
<b>1 семестр</b>				
1-3	2	Написание простейших программ с использованием операторов ввода, вывода.	6	
4-6	3	Написание программ с разветвлением.	6	
7-9	4	Написание программ с циклами.	5	
9-12	5	Написание программ с использованием строк.	8	
13-17	6	Написание программ с использованием массивов.	9	
Итого:			<b>34</b>	
<b>2 семестр</b>				
1-2	7	Написание программ с использованием структур.	2	
3-4	8	Написание программ с использованием файлов.	2	
5-7	9	Использования процедур и функций в программах.	2	
8-10	10	Написание программ с использованием рекурсий.	4	
11-13	11	Написание программ с использованием модулей.	3	
14-17	12	Динамические структуры данных.	3	
18	7-12	Защита лабораторных работ.	2	
Итого			<b>18</b>	

### Самостоятельная работа студента

Таблица 4.6

Раздел дисциплины	№ п/п	Вид самостоятельной работы студента (СРС) и перечень дидактических единиц	Трудоемкость, часов
<b>2 семестр</b>			
2	2.1	Обзор современного состояния компьютерной техники.	8
	2.2	Алгоритмы. Способы записи. Примеры.	9
3	3.1	Типы данных. Эквивалентность и тождественность типов.	17
4	4.1	Символьный тип, ограниченные и перечисляемые типы. Основные операции над простыми типами данных.	17
5	5.1	Основные базовые структуры. Структурное программирование, анализ программ, корректность программ.	22
6	6.1	Сортировка методом прямого выбора, методом парных перестановок, методом слияния	12

2-6		Подготовка к лабораторным работам	5
		Итого	<b>87</b>
<b>Зсеместр</b>			
7	7.1	Файлы прямого и последовательного доступа. Типизированные и нетипизированные файлы.	7
8	8.1	Параметры процедур: параметры-переменные, параметры-значения, константы, массивы открытого типа	8
9	9.1	Построение списков разных типов: стеки, очереди.	8
10	10.1	Понятие рекурсии. Использование рекурсии для записи решений.	10
11	11.1	Организация взаимодействия программных модулей различного типа. Использование нестандартных процедур в программах.	7
12	12.1	Оценка качества разработанных программ. Жизненный цикл программ. Основы доказательства правильности программ. Сертификация программ.	7
7-12		Подготовка к лабораторным работам	7
<b>ИТОГО:</b>			<b>54</b>

## 5 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При реализации программы дисциплины «Программирование» реализуются как традиционные технологии в виде аудиторных занятий, состоящих из лекционных (70 часов) и практических занятий (35 часов) так и компьютерные – при проведении лабораторных работ (70 часов) и тестировании остаточных знаний студентов. Самостоятельная работа студентов (109 часов) подразумевает работу под руководством преподавателя (консультация и помощь при выполнении лабораторных работ и курсовой работы), и индивидуальную работу студентов в компьютерном классе или библиотеке университета.

## 6 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ВХОДНОГО И ТЕКУЩЕГО КОНТРОЛЯ, ПРОМЕЖУТОЧНОЙ И ИТОГОВОЙ АТТЕСТАЦИИ (АННОТАЦИЯ)

*Контроль освоения дисциплины производится в соответствии с Положением о рейтинговой системе оценки знаний студентов ДИТИ НИЯУ МИФИ.*

*Текущий контроль студентов производится в дискретные временные интервалы лектором и преподавателем, ведущими лабораторные работы и практические занятия по дисциплине в следующих формах:*

- *тестирование;*
- *письменные домашние задания;*
- *выполнение лабораторных работ;*
- *защита лабораторных работ;*
- *устные опросы;*
- *контрольные работы*

*Промежуточный контроль студентов производится в следующих формах:*

- *тестирование;*
- *контрольные работы;*
- *защита лабораторных работ (тестирование);*

*Итоговый контроль по результатам семестров по дисциплине проходит в форме письменного экзамена (включает в себя ответ на теоретические вопросы и/или решения задач).*

Фонды оценочных средств, включающие типовые задания, контрольные работы, тесты и методы контроля, позволяющие оценить результаты обучения по данной дисциплине, перечислены в Приложении.

## **7 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **7.1 Перечень основной и дополнительной учебной литературы**

Таблица 7.1 - Обеспечение дисциплины основной и дополнительной литературой по дисциплине

N п/п	Автор	Название	Место издания	Наименование издательства	Год издания	Количество экземпляров
<b>Основная литература</b>						
1	Орлов С А	Теория и практика языков программирования .Учебник для вузов. Стандарт 3-го поколения		Издательство «Питер»	2013	3
	Билинг В.А.	Основы программирования на С# : учебное пособие	М.	Интернет-Университет Информационных Технологий	2010	2
	Бабалова, И.Ф	Алгоритмизация задач и структурирование программ [Электронный ресурс]	Москва	НИЯУ МИФИ	2013	
<b>Дополнительная литература</b>						
1	Марченко А. Л..	Основы программирования на С# 2.0 : учебное пособие	М.	Интернет-Университет Информационных Технологий,	2009	2
	Фридман А Л.	Язык программирования Си++. Курс лекций : учебное пособие	М.	Интернет-Университет Информационных Технологий	2010	2

### **7.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

### **7.3 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

## **8 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

1. Для проведения лекционных занятий используется:

- комплект электронных презентаций/слайдов;
- компьютерный класс, оснащенный презентационной техникой (проектор, интерактивная доска, компьютер).

2. Для проведения лабораторных работ используется:

- компьютерный класс, оснащенный презентационной техникой (проектор, интерактивная доска, компьютер);
- пакеты прикладного программного обеспечения MS Office, Visual Studio/

## 9 ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ

### Технологическая карта рейтинговых баллов по дисциплине

#### « Программирование »

Специальность 09.03.01 «Информатика и вычислительная техника»

1 курс дневное обучение

Максимальное количество баллов за работу в течение семестра: 60 баллов.

Итоговый контроль: 40 баллов

Семестр 2

Всего часов 216

в том числе:

- 1 лекции - 34 часов;
- 2 лабораторные работы - 34 часов;
- 3 семинарские / практические занятия - 34 часов;
- 4 подготовка к лекциям - \_\_\_\_\_ часов;
- 5 подготовка к семинарским / практическим занятиям - \_\_\_\_\_ часов;
- 6 подготовка к лабораторным работам - \_\_\_\_\_ часов;
- 7 подготовка к экзамену / зачету - \_\_\_\_\_ часов;
- 8 творческая самостоятельная работа (за исключением пп. 4 – 7) - \_\_\_\_\_ часов

#### Структура текущего и промежуточного контроля.

Информация о контр. точках	Текущий контроль(<=25) (ТК)									Промежуточный контроль (<=30) (ПК)		Форма итогового контроля
	ТК <sub>1</sub>	ТК <sub>2</sub>	ТК <sub>3</sub>	ТК <sub>4</sub>	ТК <sub>5</sub>	ТК <sub>6</sub>	ТК <sub>7</sub>	ТК <sub>8</sub>	ТК <sub>8</sub>	ПК <sub>1</sub>	ПК <sub>2</sub>	
форма контроля	Л/ЛБ <sub>1</sub> /ПР <sub>1</sub>	Л/ЛБ <sub>2</sub> /ПР <sub>2</sub>	Л/ЛБ <sub>3</sub> /ПР <sub>3</sub>	Л/ЛБ <sub>4</sub> /ПР <sub>4</sub>	Л/ЛБ <sub>5</sub> /ПР <sub>5</sub>	Л/ПР <sub>6</sub> ЛБ <sub>6</sub>	Л/ЛБ <sub>7</sub> /ПР <sub>7</sub>	Л/ЛБ <sub>8</sub> /ПР <sub>8</sub>	Л/ЛБ <sub>9</sub> /ПР <sub>9</sub>	КР	КР	3
неделя сдачи	2	4	6	7	10	12	13	15	18	8	14	
макс. балл	2	2	3	3	3	3	3	3	3	15	15	

#### Структура баллов, начисляемых студентам по результатам текущего контроля (промежуточного контроля)

№ п/п	Наименование видов учебной работы	Начисляемое количество баллов (долей баллов)	Максимальное количество баллов по данному виду учебной

			работы
1.	Посещение лекций.	18 лекций по 0,5 балла	9
2.	Посещение лабораторных занятий и выполнение лабораторной работы	9 работ по 0,5 балла	4,5
3.	Посещение практических занятий и выполнение заданий на практических занятиях	9 работ по 0,5 балла 7 работ по 1 баллу	11,5
<i>Максимальная сумма баллов по результатам текущего контроля</i>			25

### ПЕРЕЧЕНЬ домашних заданий и видов самостоятельной работы студентов

№ п/п	Темы домашних заданий и самостоятельной работы	Недели семестра, в которых будет выдаваться задание	Недели семестров, в которых будут приниматься отчеты по домашним заданиям и работам
1.	Построение схем ветвящихся и циклических алгоритмов. (ЛР№1, ПР№1)	2	2
2.	Построение сложных многостраничных алгоритмов. (ЛР№2 ПР№2-3)	3	4
3.	Программирование условных операторов и переключателей. (ЛР№3 ПР№4-5)	5	6
4.	Программирование циклов. (ЛР№4 ПР№6-7)	6	7
5.	Программирование циклов. (ЛР№5 ПР№8-9)	9	10
6.	Работа с одномерными и двумерными массивами. (ЛР№7 ПР№12-13)	12	13

### Технологическая карта рейтинговых баллов по дисциплине

#### « Программирование »

*Специальность 09.03.01 «Информатика и вычислительная техника»*

*1 курс дневное обучение*

Максимальное количество баллов за работу в течение семестра: 60 баллов.

Итоговый контроль: 40 баллов

Семестр 3

Всего часов 144

в том числе:

1 лекции - 18 часов;

2 лабораторные работы - 18 часов;

3 семинарские / практические занятия - 18 часов;

- 4 подготовка к лекциям - \_\_\_\_\_ часов;  
 5 подготовка к семинарским / практическим занятиям - \_\_\_\_\_ часов;  
 6 подготовка к лабораторным работам - \_\_\_\_\_ часов;  
 7 подготовка к экзамену / зачету - \_\_\_\_\_ часов;  
 8 творческая самостоятельная работа (за исключением пп. 4 – 7) - \_\_\_\_\_ часов

Структура текущего и промежуточного контроля.

Информация о контр. точках	Текущий контроль(<=25) (ТК)									Промежуточный контроль (<=30) (ПК)		Форма итогового контроля
	ТК <sub>1</sub>	ТК <sub>2</sub>	ТК <sub>3</sub>	ТК <sub>4</sub>	ТК <sub>5</sub>	ТК <sub>6</sub>	ТК <sub>7</sub>	ТК <sub>8</sub>	ТК <sub>8</sub>	ПК <sub>1</sub>	ПК <sub>2</sub>	
форма контроля	Л/ЛБ <sub>1</sub> /ПР <sub>1</sub>	Л/ЛБ <sub>2</sub> /ПР <sub>2</sub>	Л/ЛБ <sub>3</sub> /ПР <sub>3</sub>	Л/ЛБ <sub>4</sub> /ПР <sub>4</sub>	Л/ЛБ <sub>5</sub> /ПР <sub>5</sub>	Л/ПР <sub>6</sub> ЛБ <sub>6</sub>	Л/ЛБ <sub>7</sub> /ПР <sub>7</sub>	Л/ЛБ <sub>8</sub> /ПР <sub>8</sub>	Л/ЛБ <sub>9</sub> /ПР <sub>9</sub>	КР	КР	3
неделя сдачи	2	4	6	7	10	12	13	15	17	8	14	
макс. балл	2	2	3	3	3	3	3	3	3	15	15	40

Структура баллов, начисляемых студентам по результатам текущего контроля (промежуточного контроля)

№ п/п	Наименование видов учебной работы	Начисляемое количество баллов (долей баллов)	Максимальное количество баллов по данному виду учебной работы
1.	Посещение лекций.	18 лекций по 0,5 балла	9
2.	Посещение лабораторных занятий и выполнение лабораторной работы	9 работ по 0,5 балла	4,5
3.	Посещение практических занятий и выполнение заданий на практических занятиях	9 работ по 0,5 балла 7 работ по 1 баллу	11,5
<b>Максимальная сумма баллов по результатам текущего контроля</b>			<b>25</b>

ПЕРЕЧЕНЬ домашних заданий и видов самостоятельной работы студентов

№ п/п	Темы домашних заданий и самостоятельной работы	Недели семестра, в которых будет выдаваться задание	Недели семестров, в которых будут приниматься отчеты по домашним заданиям и работам
1.	Написание программ с использованием структур.	1	2
2.	Написание программ с использованием файлов.	3	4

3.	Использования процедур и функций в программах.	5	6
4.	Написание программ с использованием рекурсий.	7	8
5.	Написание программ с использованием модулей.	10	15
6.	Динамические структуры данных.	14	17



### Аннотация рабочей программы

Дисциплина **Программирование** является базовой частью профессионального модуля дисциплин подготовки студентов по направлению подготовки 09.03.01 Информатика и вычислительная техника. Дисциплина реализуется на информационно-технологическом факультете ДИТИ НИЯУ МИФИ кафедрой информационных технологий.

Дисциплина нацелена на формирование профессиональных компетенций: ПК-3- способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования ПК-4 Способность осуществлять организацию и управление проектами в области информатики и вычислительной техники в соответствии с действующими правовыми нормами и требованиями заказчика; ОПК-9- способность осваивать методики использования программных средств для решения практических задач; дополнительные компетенции: ДК-3- способность решать задачи производственной и технологической деятельности на профессиональном уровне, включая разработку алгоритмических решений в области прикладного программирования; ДК-7- использовать ЭВМ для управления и обработки информации

Целью дисциплины является обучение студентов навыкам создания программ с использованием современных технологий структурного и модульного программирования, освоение и получение навыков работы на современных вычислительных средствах, отладки и тестирования программ.

В задачи дисциплины входят: ознакомление студентов с основными компьютерными технологиями (языками, библиотеками, инструментами), используемыми для решения прикладных задач; формирование у студентов необходимого объема знаний о прикладном программировании и вычислительных методах; овладение основами постановки прикладных задач, их функционального и объектно-ориентированного анализа; овладение основами проектирования прикладного программного обеспечения, его разработки, отладки и тестирования; получение студентами практических навыков по разработке прикладных программ на языке программирования.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, практические занятия, самостоятельная работа студента, консультации, курсовое проектирование.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости в форме выполнения лабораторных работ, выполнения заданий на практических занятиях и выполнение самостоятельной работы, промежуточный контроль в форме контрольной работы и итоговый контроль в форме экзамена.

Общая трудоемкость освоения дисциплины составляет 10 зачетных единиц, 360 часов. Программой дисциплины предусмотрены лекционные 54 часов, практические 54 часов, лабораторные занятия 54 часов и 141 час самостоятельной работы студента.

**Методические указания для самостоятельной работы обучающихся**

Начинать подготовку к лабораторным занятиям необходимо с изучения рекомендованной литературы. В процессе этой работы студент должен стремиться понять и запомнить основные положения рассматриваемого материала. Заканчивать подготовку следует составлением плана (конспекта) по изучаемому материалу. Это позволяет составить концентрированное, сжатое представление по изучаемым вопросам.

В процессе подготовки к занятиям рекомендуется взаимное обсуждение материала в студенческой среде, во время которого закрепляются знания, а также приобретается практика в изложении и разъяснении полученных знаний, развивается речь.

При необходимости, студенту необходимо обращаться за консультацией к преподавателю. Идя на консультацию, необходимо хорошо продумать вопросы, которые требуют разъяснения.

**Методические указания для студентов по освоению дисциплины**

Трудоемкость освоения дисциплины составляет 360 часов, из них 108 часов аудиторных занятий и 141 час, отведенных на самостоятельную работу студента.

вид учебных занятий	Организация деятельности студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии. Уделить внимание следующим понятиям ( <i>цикл, массив, записи, файлы, динамическая память</i> ) и др.
Практические занятия	Проработка рабочей программы, уделяя особое внимание целям и задачам структуре и содержанию дисциплины. Конспектирование источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с текстом. Прослушивание аудио- и видеозаписей по заданной теме, решение расчетно-графических заданий, решение задач по алгоритму и др.
Контрольная работа/индивидуальные задания	Знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, терминов, сведений, требующих для запоминания и являющихся основополагающими в этой теме. Составление аннотаций к прочитанным литературным источникам и др.
курсовая работа	<i>Курсовая работа:</i> изучение научной, учебной, нормативной и другой литературы. Отбор необходимого материала; формирование выводов и разработка конкретных рекомендаций по решению поставленной цели и задачи; проведение практических исследований по данной теме. Инструкция по выполнению требований к оформлению курсовой работы находится в методических материалах по дисциплине.
Практикум / лабораторная работа	Методические указания по выполнению лабораторных работ
Подготовка к экзамену	При подготовке к экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и др.

## ТЕХНОЛОГИИ И ФОРМЫ ПРЕПОДАВАНИЯ

### Рекомендации по организации и технологиям обучения для преподавателя

#### I. Образовательные технологии

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

**Информационные технологии:** использование электронных образовательных ресурсов при подготовке к лекциям, практическим и лабораторным занятиям.

**Работа в команде:** совместная работа студентов в группе при выполнении лабораторных работ, выполнении групповых домашних заданий по разделу.

#### II. Виды и содержание учебных занятий

##### 2 семестр

**Теоретические занятия (лекции) - 34 часа.**

**Лекция 1.** *Информационная лекция.* Введение в программирование и языки. Обзор современного состояния компьютерной техники.

1.1. Введение в программирование и языки. Обзор современного состояния ПК и характеристик ЭВМ. Характеристики ЭВМ и её компонент. Единицы информации. Операционные системы, основные характеристики. Дисковые операционные системы (DOS), многооконные операционные системы (Windows). Понятие файла в операционной системе. Команды в операционных системах.

**Лекция 2-4.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Основные этапы решения задач на ЭВМ. Алгоритмы. Способы записи алгоритма.

2.1. Алгоритмы и программы. Основные правила разработки алгоритмов для решения задач. Базовые алгоритмические структуры: следование, развилка, повторение. Способы их изображения. Примеры алгоритмов.

**Лекция 5-7.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Обзор современных языков программирования. Данные и программы. Критерии качества программы.

3.1. Сравнительная характеристика языков программирования. Синтаксис, семантика. Общая характеристика языков программирования. Сравнение развития языков в представлении данных и способах реализации алгоритмов. Представление основных управляющих структур программирования

**Лекция 8-10.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Структура программы на языке высокого уровня. Стандартные типы данных.

4.1. Основные этапы разработки программ. Структура языковой программы. Средства для написания программ: алфавит, лексика, синтаксис языка. Термины языка. Система типов данных. Стандартные и скалярные типы данных. Диапазоны изменения различных типов данных. Символьный тип, ограниченные и перечисляемые типы. Основные операции над простыми типами данных.

4.2. Константы в языке. Простые, типизированные, адресные, структурированные константы. Определение переменных. Преобразование типов данных. Примеры использования рассмотренных структур. Выражение в языке для разных типов данных.

**Лекция 11-13.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Операторы языка : присваивания, ветвление, повторение, итерация.

5.1. Операторы языка: присваивания, ввод, вывод. Простейшие программы. Организация повторений работы программы. Директивы компилятора для обработки программ. Составной и условный операторы.

5.2. Оператор выбора. Работа со строками. Внутреннее представление строк. Простейшие циклы. Циклы с итерациями и циклы в циклах. Примеры использования языковых конструкций.

**Лекция 14-18.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Типы данных, определяемые пользователем: массивы, структуры, объединения, файлы.

6.1. Структуры данных и алгоритмы. Массив как структура данных. Сортировка массивов.

6.2. Структуры, объединения.

6.3. Организация работы с файлами. Типы файлов. Примеры.

**Практические и семинарские занятия - 34 часов.**

**Занятие 1.** Разработка алгоритмов.

**Занятие 2-3.** Разработка простейших программ.

**Занятие 3.** Разработка программ с разветвлением.

**Занятие 4.** Разработка программ с циклами.

**Занятие 4-5.** Работа со строками.

**Занятие 6-8.** Работа с массивами.

**Лабораторный практикум - 34 часов, 5 работ.**

**Лабораторная работа 1.** Написание простейших программ с использованием операторов ввода, вывода.

**Лабораторная работа 2.** Написание программ с разветвлением.

**Лабораторная работа 3.** Написание программ с циклами.

**Лабораторная работа 4.** Написание программ с использованием строк.

**Лабораторная работа 5.** Написание программ с использованием массивов.

**Управление самостоятельной работой студента.**

**2 семестр**

**Теоретические занятия (лекции) - 18 часов.**

**Лекция 1-2.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Способы конструирования программ. Процедуры и функции.

7.1. Процедуры и функции в языке. Основные понятия. Принципы использования процедур и функций в программах. Параметры процедур и функций.

**Лекция 3-4.** *Проблемная лекция. Опрос по теме предыдущей лекции.* Разработка простейшего интерфейса. Диалоговые программы.

8.1. Проектирование программ. Постановка задачи и спецификация программы. Принципы структурного программирования. Этапы разработки программ. Организация интерфейса.

**Лекция 5-7.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Динамические структуры данных. Списки: основные виды и способы.

9.1. Динамические структуры данных. Указатели и ссылки. Динамические массивы. Построение списков разных типов: стеки, очереди.

**Лекция 8-11.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Сортировка массивов и списков

**Лекция 12-15.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Программирование рекурсивных алгоритмов.

11.1. Понятие рекурсии. Использование рекурсии для записи решений. Древоподобные структуры. Бинарные деревья.

**Лекция 16-17.** *Лекция-беседа. Опрос по теме предыдущей лекции.* Способы конструирования программ. Модульные программы. Основы доказательства правильности.

12.1 Модули в языке: назначение, структура, трансляция, тестирование. Особенности использования модулей. Стандартные модули.

12.2. Понятие дружественного интерфейса. Создание элементов меню. Использование шаблонов меню. Оценка качества разработанных программ. Основы доказательства правильности программ.

**Практические и семинарские занятия - 18 часов.**

**Занятие 1.** Работа со структурами.

**Занятие 2-3.** Работа с файлами.

**Занятие 4-5.** Сортировка массивов.

**Занятие 5-6.** Принципы использования процедур и функций в программах.

Занятие 6-7. Использование рекурсии для записи решений.

Занятие 7-8. Динамические структуры данных. Модули.

**Лабораторный практикум - 18 часа, 6 работ.**

**Лабораторная работа 1.** Написание программ с использованием структур

**Лабораторная работа 2.** Написание программ с использованием файлов.

**Лабораторная работа 3.** Использование процедур и функций в программах.

**Лабораторная работа 4.** Написание программ с использованием рекурсий.

**Лабораторная работа 5.** Написание программ с использованием модулей.

**Лабораторная работа 6.** Динамические структуры данных.

**Управление самостоятельной работой студента.**

*Консультации по выполнению курсовой работы.*

*Проверка готовности к лабораторной работе.*