

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агафонов Александр Викторович

Должность: Директор филиала

Дата подписания: 18.06.2023 09:54:58

Удостоверение: 18062023095458

2559477a8ec1706dc9cf164bc411eb6d5c4ab06

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**  
**ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**  
**ЧЕБОКСАРСКИЙ ИНСТИТУТ (ФИЛИАЛ) МОСКОВСКОГО ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА**

## Кафедра Информационных технологий и систем управления



**УТВЕРЖДАЮ**

Директор филиала

А.В. Агафонов

"27" мая 2026г.

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

## «Программирование и основы алгоритмизации»

(наименование дисциплины)

Направление подготовки	<b>09.03.02 «Информационные системы и технологии»</b> (код и наименование направления подготовки)
Направленность (профиль) подготовки	<b>«Информационные технологии в медиаиндустрии и дизайне»</b> (наименование профиля подготовки)
Квалификация выпускника	<b>бакалавр</b>
Форма обучения	<b>очная, заочная</b>
Год начала обучения	<b>2026</b>

Чебоксары, 2026

Рабочая программа дисциплины разработана в соответствии с:

- федеральным государственным образовательным стандартом высшего образования - бакалавриат по направлению подготовки 09.03.02 – Информационные системы и технологии, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 926 от 19 сентября 2017 г. зарегистрированный в Минюсте 12 октября 2017 года, рег. номер 48535 (далее – ФГОС ВО);
- учебным планом (очной, заочной форм обучения) по направлению подготовки 09.03.02 «Информационные системы и технологии».

Рабочая программа дисциплины включает в себя оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине (п.6 Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины).

Автор Пикина Наталия Евгеньевна, кандидат педагогических наук, доцент кафедры информационных технологий и систем управления

(указать ФИО, ученую степень, ученое звание или должность)

Программа одобрена на заседании кафедры Информационных технологий и систем управления (протокол № 9 от 22.05.2026 г.).

**1. Перечень планируемых результатов обучения, соотнесенных с планируемыми результатами освоения образовательной программы (Цели освоения дисциплины)**

1.1. *Целями* освоения дисциплины «Программирование и основы алгоритмизации» являются:

формирование у студентов знаний об основных принципах алгоритмизации и теории алгоритмов, программе и программировании.

Для достижения целей дисциплины необходимо решить следующую *основную задачу* – сформировать у обучающихся теоретические знания и практические навыки, необходимые для:

формирования практических навыков создания прикладных программных продуктов на основе современных технологий программирования с использованием современных алгоритмических языков (C/C++, Python и др.).

1.2. Области профессиональной деятельности и (или) сферы профессиональной деятельности, в которых выпускники, освоившие программу, могут осуществлять профессиональную деятельность:

*Об Связь, информационные и коммуникационные технологии (в сфере проектирования, разработки, внедрения и эксплуатации средств вычислительной техники и информационных систем, управления их жизненным циклом).*

1.3. К основным задачам изучения дисциплины относится подготовка обучающихся к выполнению трудовых функций в соответствии с профессиональными стандартами:

<b>Наименование профессиональных стандартов (ПС)</b>	<b>Код, наименование и уровень квалификации ОТФ, на которые ориентирована дисциплина</b>	<b>Код и наименование трудовых функций, на которые ориентирована дисциплина</b>
06.025 Профессиональный стандарт «Специалист по дизайну графических пользовательских интерфейсов», утв. приказом Министерством труда и социальной защиты РФ 29 сентября 2020 № 671н	Д Эвристическая оценка графического пользовательского интерфейса	D/01.6 Формальная оценка графического пользовательского интерфейса D/02.6 Анализ данных о действиях пользователей при работе с интерфейсом

Наименование профессиональных стандартов (ПС)	Код, наименование и уровень квалификации ОТФ, на которые ориентирована дисциплина	Код и наименование трудовых функций, на которые ориентирована дисциплина
06.015 Профессиональный стандарт «Специалист по информационным системам», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2014 г. № 896н (зарегистрирован Министерством юстиции Российской Федерации 24 декабря 2014 г., регистрационный № 35361)	С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	С/14.6 Разработка архитектуры ИС С/15.6 Разработка прототипов ИС С/16.6 Проектирование и дизайн ИС С/18.6 Организационное и технологическое обеспечение кодирования на языках программирования

#### 1.4. Компетенции обучающегося, формируемые в результате освоения дисциплины

Наименование категории (группы) компетенций	Код и наименование компетенций	Код и наименование индикатора достижения компетенции	Перечень планируемых результатов обучения
Разработка и реализация проектов	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение	<i>на уровне знаний:</i> знать этапы жизненного цикла проекта; знать этапы разработки и реализации проекта; знать методы разработки и управления проектами <i>на уровне умений:</i> уметь разрабатывать проект с учетом анализа альтернативных вариантов его реализации, определять с использованием инструментов планирования целевые этапы и основные направления работ; уметь формулировать цель задачи, обосновывать актуальность, научную и практическую значимость, ожидаемые результаты и возможные сферы применения; <i>на уровне навыков:</i> Владеть управлению

		<p>УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации</p> <p>УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования</p>	<p>жизненным циклом продукции и ее качеством, в мероприятиях по контролю соответствия разрабатываемых проектов и технической документации, действующим стандартам</p> <p>-----</p> <p>-</p> <p><i>на уровне знаний:</i>  знать технические характеристики и экономические показатели отечественных и зарубежных образцов программно-технических комплексов систем автоматизации и управления</p> <p><i>на уровне умений:</i>  уметь управлять проектом на всех этапах его жизненного цикла, осуществлять мониторинг хода его реализации, вносить при необходимости изменения в план реализации проекта;  уметь представлять публично результаты проекта (его этапов) в различной форме (отчеты, статьи, выступления на научно-практических конференциях, семинарах)</p> <p><i>на уровне навыков:</i>  навыками разработки (на основе действующих стандартов и другой нормативной документации) проектной и рабочей технической документации в области систем автоматизации и управления, их эксплуатационному обслуживанию</p> <p>-----</p> <p>-</p> <p><i>на уровне знаний:</i>  знать современные тенденции развития средств и систем автоматизации и управления, средств вычислительной техники, коммуникаций и связи,</p> <p><i>на уровне умений:</i>  уметь использовать при разработке проектной и рабочей документации на системы автоматизации и управления</p>
--	--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

			действующие стандарты, технические условия и другие нормативные документы. <i>на уровне навыков:</i> методиками разработки и управления проектом; методами оценки потребности в ресурсах и эффективности проекта
Алгоритмизация и программирование	ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ОПК-6.1. Знать: методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.	<i>на уровне знаний:</i> знать стандарты, методические и нормативные материалы, сопровождающие проектирование, изготовление, внедрение и эксплуатацию средств и систем автоматизации и управления. <i>на уровне умений:</i> уметь использовать при разработке проектной и рабочей документации на системы автоматизации и управления действующие стандарты, технические условия и другие нормативные документы. <i>на уровне навыков:</i> навыками применения современных программных средств выполнения и редактирования изображений и чертежей и подготовки конструкторско-технологической документации при автоматизации управления.
		Знать: методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.	<i>на уровне знаний:</i> знать методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий. <i>на уровне умений:</i> уметь применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий <i>на уровне навыков:</i> навыками программирования, отладки и тестирования
		ОПК-6.3. Иметь навыки: программирования,	<i>на уровне знаний:</i> знать современные информационные технологии,

		отладки и тестирования прототипов программно-технических комплексов задач.	методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности уметь использовать информационные технологии и информационно-вычислительные системы для решения научно-исследовательских и проектных задач систем управления <i>на уровне навыков:</i> навыками выбора наиболее оптимальных прикладных программных пакетов для решения соответствующих задач
--	--	----------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2. Место дисциплины в структуре ОПОП

Дисциплина Б1.Д(М).Б.21 «Программирование и основы алгоритмизации» реализуется в рамках обязательной части Блока 1 «Дисциплины (модули)» программы бакалавриата.

Дисциплина «Программирование и основы алгоритмизации» преподается обучающимся по очной форме обучения – во 2-м и 3-м семестрах, по заочной форме – во 3-м и 4-м семестрах.

Дисциплина «Программирование и основы алгоритмизации» является промежуточным этапом формирования компетенций УК-2, ОПК-6 процессе освоения ОПОП.

Дисциплина «Программирование и основы алгоритмизации» основывается на знаниях, умениях и навыках, приобретенных при изучении дисциплин Введение в проектную деятельность, Инженерная и компьютерная графика и является предшествующей для изучения дисциплин Проектная деятельность, Учебная практика: ознакомительная практика, Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-технологическая) практика, Метрология, стандартизация и сертификация, Производственная практика: технологическая (проектно-технологическая) практика, Производственная практика: научно-исследовательская работа, Производственная практика: преддипломная практика, Государственная итоговая аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы.

Формой промежуточной аттестации знаний обучаемых по очной форме обучения является зачет во 2-м семестре, курсовая работа и экзамен в 3-м семестре, по заочной форме зачет в 3-м семестре, курсовая работа и экзамен во 4-м семестре.

### 3. Объем дисциплины

очная форма обучения:

Вид учебной работы по дисциплине	Всего в з.е. и часах	Семестр 2 в часах	Семестр 3 в часах
<b>Общая трудоёмкость дисциплины</b>	<b>7 з.е. - 252 ак.час</b>	<b>72 ак.час</b>	<b>180 ак.час</b>
<b>Контактная работа - Аудиторные занятия</b>	<b>101</b>	<b>36</b>	<b>65</b>
Лекции	50	18	32
Лабораторные занятия	50	18	32
Семинары, практические занятия	-	-	-
Консультация	1	-	1
<b>Самостоятельная работа</b>	<b>115</b>	<b>36</b>	<b>79</b>
Курсовая работа (курсовой проект)	2	-	2
Вид промежуточной аттестации	Экзамен – 36 часов	Зачет	Экзамен – 36 часов

заочная форма обучения:

Вид учебной работы по дисциплине	Всего в з.е. и часах	Семестр 3 в часах	Семестр 4 в часах
<b>Общая трудоёмкость дисциплины</b>	<b>7 з.е. - 252 ак.час</b>	<b>72 ак.час</b>	<b>180 ак.час</b>
<b>Контактная работа - Аудиторные занятия</b>	<b>21</b>	<b>8</b>	<b>13</b>
Лекции	10	4	6
Лабораторные занятия	10	4	6
Семинары, практические занятия	-	-	-
Консультация	1	-	1
<b>Самостоятельная работа</b>	<b>218</b>	<b>60</b>	<b>158</b>
Курсовая работа (курсовой проект)	2	-	2
Вид промежуточной аттестации	Экзамен – 13 часов	Зачет – 4 часа	Экзамен – 9 часов

**4. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий**

#### 4.1. Учебно-тематический план

Очная форма обучения

Наименование тем (разделов) дисциплины	Трудоёмкость в часах				Код индикатора достижений компетенции
	Контактная работа – Аудиторная работа			самостоятельная работа	
	лекции	лабораторные занятия	семинары и практические занятия		
Тема1. Введение в языки программирования.	6	6	-	14	УК-2.1, УК-2.2, УК-2.3, ОПК-6.1,

Типы данных в языках Си.					ОПК-6.2, ОПК-6.3
Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции	6	6	-	14	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 3. Массивы. Указатели. Динамические переменные. Функции	6	6	-	14	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 4. Сценарные языки. Язык Python.	6	6	-	14	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой	6	6	-	14	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.	6	6	-	14	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 7. Элементы функционального программирования на языке Python	6	6	-	14	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 8. Модульное проектирование. Графический интерфейс на Python.	8	8	-	15	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Консультация		1		-	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Курсовая работа (курсовой проект)			2		УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Контроль (экзамен)			36		УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
<b>ИТОГО</b>		<b>103</b>		<b>113</b>	

### Заочная форма обучения

Наименование тем (разделов) дисциплины	Трудоемкость в часах				Код индикатора достижений компетенции
	Контактная работа – Аудиторная работа			самостоятельная работа	
	лекции	лабораторные занятия	семинары и практические занятия		

Тема 1. Введение в язык программирования. Типы данных в языках Си.	2	-	-	26	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции	2	-	-	26	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 3. Массивы. Указатели. Динамические переменные. Функции	2	-	-	26	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 4. Сценарные языки. Язык Python.	2	2	-	28	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой	2	2	-	28	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.	-	2	-	28	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 7. Элементы функционального программирования на языке Python	-	2	-	28	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Тема 8. Модульное проектирование. Графический интерфейс на Python.	-	2	-	28	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Консультация		1		-	УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Курсовая работа (курсовой проект)			2		УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
Контроль (экзамен)			13		УК-2.1, УК-2.2, УК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3
<b>ИТОГО</b>		<b>21</b>		<b>218</b>	

#### 4.2. Содержание дисциплины

## **Тема 1. Введение в языки программирования. Типы данных в языках Си.**

История и классификация языков программирования. Компилируемые и интерпретируемые языки.

Структура программы на языке Си. Основные элементы: заголовки, функции, операторы.

Типы данных: целые, вещественные, символьные. Размеры и диапазоны.

Переменные и константы. Инициализация и область действия.

Простейший ввод/вывод данных. Функции printf и scanf.

## **Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции**

Арифметические, логические и побитовые операции. Приоритеты операторов.

Условные конструкции: if, else, switch-case.

Циклические конструкции: for, while, do-while. Операторы управления break, continue.

Понятие блок-схем и их связь с программной реализацией.

Практика разработки простейших алгоритмов.

## **Тема 3. Массивы. Указатели. Динамические переменные. Функции**

Одномерные и многомерные массивы. Индексация, инициализация, обход.

Основы работы с указателями. Адресная арифметика.

Связь массивов и указателей. Строки как массивы символов.

Выделение памяти: malloc, calloc, free. Работа с динамическими структурами данных.

Объявление и вызов функций. Возврат значений и передача параметров по значению и по ссылке.

## **Тема 4. Сценарные языки. Язык Python.**

Отличия сценарных языков от системных. Преимущества Python.

Установка и запуск Python. Использование интерактивной оболочки (REPL).

Синтаксис Python. Отступы как средство структуры.

Сравнение синтаксиса Python и Си на простых примерах.

## **Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой**

Основные типы: int, float, str, bool. Операции над ними.

Строки: индексация, срезы, методы строк.

Списки и их методы. Вложенные списки. Изменяемость объектов.

Логические выражения и операторы if, elif, else.

Циклы for и while в Python. Итерируемые объекты.

## **Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.**

Объявление и вызов функций. Аргументы по умолчанию и именованные параметры.

Возврат значений. Функции как объекты.

Локальная и глобальная область видимости. Ключевые слова `global` и `nonlocal`.

Параметры: передача по значению и по ссылке (поведение изменяемых объектов).

## **Тема 7. Элементы функционального программирования на языке Python**

Лямбда-выражения и анонимные функции.

Встроенные функции `map`, `filter`, `reduce`.

Функции высшего порядка. Замыкания.

Введение в рекурсию. Примеры рекурсивных функций.

Итераторы и генераторы. Использование `yield`.

## **Тема 8. Модульное проектирование. Графический интерфейс на Python.**

Структура проекта. Модули и пакеты. Импорт и повторное использование кода.

Принципы модульности и повторного использования.

Обзор библиотек для GUI: `tkinter`, `PyQt`, `Kivy`.

Создание простейшего графического интерфейса: окна, кнопки, обработка событий.

Примеры интеграции логики программы с графическим интерфейсом.

## **5. Учебно-методическое обеспечение самостоятельной работы студентов.**

Самостоятельная работа проводится с целью: систематизации и закрепления полученных теоретических знаний и практических умений обучающихся; углубления и расширения теоретических знаний студентов; формирования умений использовать нормативную, правовую, справочную документацию, учебную и специальную литературу; развития познавательных способностей и активности обучающихся: творческой инициативы, самостоятельности, ответственности, организованности; формирование самостоятельности мышления, способностей к саморазвитию, совершенствованию и самоорганизации; формирования профессиональных компетенций; развитию исследовательских умений студентов.

Формы и виды самостоятельной работы студентов: чтение основной и дополнительной литературы – самостоятельное изучение материала по рекомендуемым литературным источникам; работа с библиотечным каталогом, самостоятельный подбор необходимой литературы; работа со словарем, справочником; поиск необходимой информации в сети Интернет; конспектирование источников; реферирование источников; составление аннотаций к прочитанным литературным источникам; составление рецензий и отзывов на прочитанный материал; составление обзора публикаций по теме; составление и разработка терминологического словаря; составление хронологической таблицы; составление библиографии (библиографической картотеки); подготовка к различным формам текущей и промежуточной аттестации (к тестированию, контрольной работе, зачету, экзамену);

выполнение домашних контрольных работ; самостоятельное выполнение практических заданий репродуктивного типа (ответы на вопросы, задачи, тесты; выполнение творческих заданий).

Технология организации самостоятельной работы обучающихся включает использование информационных и материально-технических ресурсов образовательного учреждения: библиотеку с читальным залом, компьютерные классы с возможностью работы в Интернет; аудитории (классы) для консультационной деятельности.

Перед выполнением обучающимися внеаудиторной самостоятельной работы преподаватель проводит консультирование по выполнению задания, который включает цель задания, его содержания, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. Во время выполнения обучающимися внеаудиторной самостоятельной работы и при необходимости преподаватель может проводить индивидуальные и групповые консультации.

Контроль самостоятельной работы студентов предусматривает: соотнесение содержания контроля с целями обучения; объективность контроля; валидность контроля (соответствие предъявляемых заданий тому, что предполагается проверить); дифференциацию контрольно-измерительных материалов.

Формы контроля самостоятельной работы: просмотр и проверка выполнения самостоятельной работы преподавателем; организация самопроверки, взаимопроверки выполненного задания в группе; обсуждение результатов выполненной работы на занятии; проведение письменного опроса; проведение устного опроса; организация и проведение индивидуального собеседования; организация и проведение собеседования с группой.

### **Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы**

<b>Наименование тем (разделов) дисциплины</b>	<b>Перечень вопросов, отводимых на самостоятельное освоение</b>	<b>Формы внеаудиторной самостоятельной работы</b>
Тема 1. Введение в языки программирования. Типы данных в языках Си.	<ol style="list-style-type: none"> <li>1. История и развитие языков программирования: от ассемблера до современных языков.</li> <li>2. Основные типы данных в языке Си: целые, вещественные, символы и логические.</li> <li>3. Структуры данных: массивы, структуры, перечисления.</li> <li>4. Понятие переменной и её роль в программировании.</li> <li>5. Область видимости переменных: локальные и глобальные переменные.</li> <li>6. Преобразование типов данных: неявное и явное преобразование.</li> <li>7. Введение в компиляцию и интерпретацию программ.</li> <li>8. Роль стандартной библиотеки в языке Си.</li> </ol>	Изучение литературы по языкам программирования и их типам данных. Выполнение практических задач на определение и использование переменных.

	<p>9. Основные операторы и их применение в языке Си. 10. Будущее языков программирования: тенденции и новые подходы.</p>	
<p>Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции</p>	<p>1. Арифметические, логические и побитовые операции в языке Си. 2. Условные операторы: if, switch и их применение. 3. Циклические конструкции: for, while, do-while. 4. Вложенные операторы: как они работают и когда их использовать. 5. Алгоритмы сортировки: простые алгоритмы и их реализация на языке Си. 6. Принципы построения алгоритмов: последовательность, ветвление, циклы. 7. Дебаггинг: методы отладки программ в языке Си. 8. Использование комментариев и документации в программировании. 9. Эффективность алгоритмов: сложность и производительность. 10. Введение в тестирование программного обеспечения.</p>	<p>Решение задач на использование циклов и условных операторов. Проведение анализа алгоритмов сортировки и их реализации.</p>
<p>Тема 3. Массивы. Указатели. Динамические переменные. Функции</p>	<p>1. Определение массивов и их использование в языке Си. 2. Основы работы с указателями: объявление, инициализация и разыменование. 3. Динамическое выделение памяти: malloc, calloc, realloc и free. 4. Функции в языке Си: определение и вызов, параметры и возвращаемые значения. 5. Область видимости функций и переменных: локальные и глобальные функции. 6. Параметры функции: передача по значению и по ссылке. 7. Массивы и указатели: взаимосвязь и различия. 8. Применение указателей для работы с динамическими структурами данных. 9. Рекурсивные функции: как они работают и когда их использовать. 10. Ошибки при работе с указателями и динамической памятью.</p>	<p>Изучение литературы по массивам и указателям в языке Си. Выполнение практических задач на динамическое выделение памяти.</p>
<p>Тема 4. Сценарные языки. Язык Python.</p>	<p>1. Определение сценарных языков и их особенности. 2. Введение в язык Python: синтаксис и базовые конструкции. 3. Типы данных в Python: строки, числа, списки, кортежи, множества и словари. 4. Управляющие структуры в Python: if, for, while. 5. Функции в Python: определение, параметры и области видимости. 6. Модули и библиотеки: как использовать стандартные и сторонние библиотеки. 7. Работа с файлами в Python: чтение и запись. 8. Исключения в Python: обработка ошибок и отладка. 9. Применение Python в различных областях: от веб-разработки до анализа данных. 10. Будущее Python: новые возможности и тренды.</p>	<p>Решение задач на использование управляющих конструкций в Python. Проведение анализа примеров использования Python в реальных проектах.</p>

<p>Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой</p>	<ol style="list-style-type: none"> <li>1. Числа в Python: целые, вещественные и комплексные.</li> <li>2. Работа со строками: методы и операции.</li> <li>3. Списки в Python: создание, доступ к элементам и методы.</li> <li>4. Словари и множества: их использование и особенности.</li> <li>5. Логические операции и выражения: and, or, not.</li> <li>6. Условные операторы: использование в логических выражениях.</li> <li>7. Преобразование типов данных и работа с ними.</li> <li>8. Введение в регулярные выражения: основы и примеры.</li> <li>9. Оптимизация работы с данными: выбор подходящих структур.</li> <li>10. Примеры задач на работу с базовыми типами данных.</li> </ol>	<p>Изучение литературы по базовым типам данных в Python. Выполнение практических задач на работу с числами и строками.</p>
<p>Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.</p>	<ol style="list-style-type: none"> <li>1. Определение функций и их роль в программировании.</li> <li>2. Различия между локальными и глобальными переменными.</li> <li>3. Типы передачи параметров в функции: по значению и по ссылке.</li> <li>4. Вложенные функции и их использование.</li> <li>5. Анонимные функции (лямбда-функции) в Python.</li> <li>6. Обработка аргументов переменной длины: *args и kwargs.</li> <li>7. Документация функций: как писать и использовать docstrings.</li> <li>8. Дебаггинг функций: методы отладки и выявления ошибок.</li> <li>9. Примеры применения функций для решения задач.</li> <li>10. Роль функций в организации кода и повышении его читаемости.</li> </ol>	<p>Решение задач на передачу данных в функции. Проведение анализа примеров использования функций в реальных проектах.</p>
<p>Тема 7. Элементы функционального программирования на языке Python</p>	<ol style="list-style-type: none"> <li>1. Определение функционального программирования и его особенности.</li> <li>2. Основные концепции функционального программирования: чистые функции, высшие функции.</li> <li>3. Использование функций map, filter и reduce в Python.</li> <li>4. Лямбда-функции: применение и примеры.</li> <li>5. Применение списковых включений (list comprehensions) в функциональном программировании.</li> <li>6. Замыкания и их использование в Python.</li> <li>7. Рекурсия в функциональном программировании: как она работает.</li> <li>8. Примеры применения функционального подхода в реальных проектах.</li> <li>9. Сравнение функционального и императивного подходов.</li> <li>10. Будущее функционального программирования: новые возможности и тренды.</li> </ol>	<p>- Изучение литературы по функциональному программированию на Python. - Выполнение практических задач на использование функции map, filter и reduce. Просмотр материалов о функциональном программировании.</p>
<p>Тема 8. Модульное проектирование. Графический интерфейс на Python.</p>	<ol style="list-style-type: none"> <li>1. Определение модульного проектирования и его преимущества.</li> <li>2. Создание модулей в Python: структура и правила организации кода.</li> <li>3. Использование пакетов и библиотек в Python.</li> </ol>	<p>Создание простого графического интерфейса с использованием Tkinter. Исследование примеров использования</p>

	<p>4. Основы графического программирования: введение в GUI.</p> <p>5. Библиотеки для создания графических интерфейсов: Tkinter, PyQt, Kivy.</p> <p>6. Создание простого графического интерфейса с помощью Tkinter.</p> <p>7. Обработка событий в графических интерфейсах.</p> <p>8. Примеры проектов на основе графических интерфейсов.</p> <p>9. Советы по дизайну пользовательского интерфейса.</p> <p>10. Будущее графического программирования на Python: тенденции и новшества.</p>	<p>графических библиотек в Python.</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------

### Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер
«Хорошо»	Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера
«Удовлетворительно»	Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой самостоятельной работы

## 6. Оценочные материалы (фонд оценочных средств) для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

### 6.1. Паспорт фонда оценочных средств

№	Контролируемые разделы (темы) дисциплины	Код и наименование компетенции	Индикатор достижения компетенции	Наименование оценочного средства
1.	Тема 1. Введение в языки программирования. Типы данных в языках Си.	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм,	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами	Опрос, тест, доклад, зачет, экзамен

		имеющихся ресурсов и ограничений	его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий комплексов задач.	Опрос, тест, доклад, зачет, экзамен
2.	Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий,	Опрос, тест, доклад, зачет, экзамен

			ресурсов и ограничений, возможностей использования	
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий	Опрос, тест, доклад, зачет, экзамен
3.	Тема 3. Массивы. Указатели. Динамические переменные. Функции	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	Опрос, тест, доклад, зачет, экзамен
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов,	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.	Опрос, тест, доклад, зачет, экзамен

		лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий	
4.	Тема 4. Сценарные языки. Язык Python.	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	Опрос, тест, доклад, зачет, экзамен
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.	Опрос, тест, доклад, зачет, экзамен

			ОПК-6.3. Владеть: методами разработки технических заданий	
5.	Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	Опрос, тест, доклад, зачет, экзамен
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий	Опрос, тест, доклад, зачет, экзамен
6.	Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными	Опрос, тест, доклад, зачет, экзамен

		решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий.	Опрос, тест, доклад, зачет, экзамен
7.	Тема 7. Элементы функционального программирования на языке Python	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом	Опрос, тест, доклад, зачет, экзамен

			действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий	Опрос, тест, доклад, зачет, экзамен
8.	Тема 8. Модульное проектирование. Графический интерфейс на Python.	УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение УК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации УК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования	Опрос, тест, доклад, зачет, экзамен
		ОПК-6. Способен разрабатывать бизнес-планы и технические задания на	ОПК-6.1. Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов	Опрос, тест, доклад, зачет, экзамен

		оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	компьютерным и сетевым оборудованием. ОПК-6.2. Уметь: анализировать ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3. Владеть: методами разработки технических заданий	
--	--	-----------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

**Этапы формирования компетенций в процессе освоения ОПОП** прямо связаны с местом дисциплин в образовательной программе. Каждый этап формирования компетенции, характеризуется определенными знаниями, умениями и навыками и (или) опытом профессиональной деятельности, которые оцениваются в процессе текущего контроля успеваемости, промежуточной аттестации по дисциплине (практике) и в процессе итоговой аттестации.

Дисциплина «Программирование и основы алгоритмизации» является начальным этапом комплекса дисциплин, в ходе изучения которых у студентов формируются компетенции УК-2, ОПК-6.

Формирование компетенции УК-2 начинается с освоения дисциплин «Введение в проектную деятельность», «Инженерная и компьютерная графика».

Формирование компетенции ОПК-6 начинается с освоения дисциплины «Программирование и основы алгоритмизации».

Завершается работа по формированию у студентов указанных компетенций в ходе изучения дисциплин «Проектная деятельность», Учебная практика: ознакомительная практика, Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-технологическая) практика, «Метрология, стандартизация и сертификация», Производственная практика: технологическая (проектно-технологическая) практика, Производственная практика: научно-исследовательская работа, Производственная практика: преддипломная практика.

Итоговая оценка сформированности компетенций УК-2, ОПК-6 определяется в период Государственная итоговая аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы.

**В процессе изучения дисциплины, компетенции также формируются поэтапно.**

Основными этапами формирования УК-2, ОПК-6 при изучении дисциплины Б1.Д(М).Б.21 «Программирование и основы алгоритмизации» является последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение студентами необходимыми дескрипторами (составляющими) компетенций. Для оценки уровня сформированности компетенций в процессе изучения дисциплины предусмотрено проведение текущего контроля успеваемости по темам (разделам) дисциплины и промежуточной аттестации по дисциплине – экзамен.

## 6.2. Контрольные задания и материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

### 6.2.1. Контрольные вопросы по темам (разделам) для опроса на занятиях

Тема (раздел)	Вопросы
Тема 1. Введение в язык программирования. Типы данных в языках Си.	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Основные особенности языков программирования высокого уровня.</li> <li>2. Отличие компилируемых языков от интерпретируемых.</li> <li>3. Структура программы на языке Си.</li> <li>4. Основные типы данных в языке Си (int, float, char, etc.).</li> <li>5. Как в языке Си осуществляется определение переменных и их типизация.</li> <li>6. Размеры стандартных типов данных в языке Си на различных платформах.</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>7. Преимущества и недостатки языка Си для решения задач низкоуровневого программирования.</li> <li>8. Особенности работы с указателями и их использование для реализации динамической памяти.</li> <li>9. Создание и использование структур данных в языке Си.</li> <li>10. Работа с типами данных, зависимыми от платформы, в языке Си.</li> <li>11. Сравнение типов данных в Си с типами данных других языков программирования (например, Python, Java).</li> </ol>
Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Основные арифметические операции в языке Си.</li> <li>2. Операции сравнения и логические операции в языке Си.</li> <li>3. Операторы ввода и вывода в языке Си.</li> <li>4. Структуры выбора: if, switch.</li> <li>5. Циклические конструкции: for, while, do-while.</li> <li>6. Примеры использования базовых алгоритмов поиска и сортировки.</li> </ol> <p>ОПК-6</p>

	<ol style="list-style-type: none"> <li>7. Преимущества и недостатки различных конструкций выбора (if, switch).</li> <li>8. Проблемы, связанные с использованием вложенных циклов и операторов выбора.</li> <li>9. Оптимизация алгоритмов с использованием различных конструкций.</li> <li>10. Применение алгоритмов сортировки в реальных задачах (например, сортировка массива).</li> <li>11. Реализация поиска в отсортированном и не отсортированном массиве.</li> </ol>
<p>Тема 3. Массивы. Указатели. Динамические переменные. Функции</p>	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Определение и использование массивов в языке Си.</li> <li>2. Операции с одномерными и многомерными массивами.</li> <li>3. Основы работы с указателями в языке Си.</li> <li>4. Динамическое выделение памяти с использованием malloc, calloc, free.</li> <li>5. Функции в языке Си: определение и вызов.</li> <li>6. Передача параметров в функции (по значению и по ссылке).</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>7. Разница между статическими и динамическими массивами.</li> <li>8. Риски и ошибки, связанные с использованием указателей (например, "утечки" памяти).</li> <li>9. Применение динамической памяти в разработке эффективных программ.</li> <li>10. Передача указателей в функции и работа с ними.</li> <li>11. Рекурсивные функции и их особенности в языке Си.</li> </ol>
<p>Тема 4. Сценарные языки. Язык Python.</p>	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Основные отличия сценарных языков от языков программирования общего назначения.</li> <li>2. Принципы работы с Python как сценарным языком.</li> <li>3. Простейшие операции в Python (арифметические операции, операции со строками).</li> <li>4. Основы ввода/вывода в Python (функции input(), print()).</li> <li>5. Создание и использование переменных в Python.</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>6. Преимущества и недостатки Python как языка программирования для начинающих.</li> <li>7. Работа с коллекциями данных (списки, множества, кортежи, словари).</li> <li>8. Основные библиотеки и фреймворки Python для решения задач.</li> <li>9. Операторы и конструкции управления потоком в Python.</li> <li>10. Мощности Python для решения задач обработки данных и автоматизации.</li> </ol>

<p>Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой</p>	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Основные типы данных в Python (int, float, str, bool).</li> <li>2. Операции над числами и строками в Python.</li> <li>3. Создание и работа со списками (списки, кортежи, множества).</li> <li>4. Использование логических операторов и условий.</li> <li>5. Основы работы со строками в Python: индексация, нарезка, методы.</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>6. Манипуляции с числами и строками в Python (форматирование строк, работа с плавающей точкой).</li> <li>7. Особенности работы с большими числами и строками в Python.</li> <li>8. Продвинутое операции с коллекциями данных (сортировка, фильтрация, агрегация).</li> <li>9. Использование регулярных выражений для обработки строк.</li> <li>10. Реализация алгоритмов обработки строк и чисел.</li> </ol>
<p>Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.</p>	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Определение и вызов функций в Python.</li> <li>2. Параметры функции и их типы.</li> <li>3. Локальные и глобальные переменные в Python.</li> <li>4. Основы области видимости переменных (scope).</li> <li>5. Передача данных в функции (по значению и по ссылке).</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>6. Рекурсия и её применение в Python.</li> <li>7. Замыкания и лямбда-функции.</li> <li>8. Разница между изменяемыми и неизменяемыми типами данных при передаче в функцию.</li> <li>9. Применение функций высшего порядка.</li> <li>10. Обработка ошибок при вызове функций.</li> </ol>
<p>Тема 7. Элементы функционального программирования на языке Python</p>	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Основы функционального программирования.</li> <li>2. Использование лямбда-функций в Python.</li> <li>3. Применение функций map(), filter(), reduce().</li> <li>4. Основы работы с анонимными функциями.</li> <li>5. Роль и возможности списковых выражений (list comprehensions) в Python.</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>6. Применение функциональных конструкций для оптимизации кода.</li> <li>7. Различия между императивным и функциональным стилем программирования.</li> <li>8. Использование функций высшего порядка для работы с коллекциями.</li> <li>9. Преимущества и недостатки функционального подхода в Python.</li> </ol>

	10. Реализация композиции функций и отложенных вычислений.
Тема 8. Модульное проектирование. Графический интерфейс на Python.	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. Основы модульного проектирования в Python.</li> <li>2. Работа с модулями и пакетами.</li> <li>3. Основы создания графических интерфейсов с использованием библиотеки Tkinter.</li> <li>4. Создание простого графического интерфейса для приложения.</li> <li>5. Обработка событий и взаимодействие с пользователем в графическом интерфейсе.</li> </ol> <p>ОПК-6</p> <ol style="list-style-type: none"> <li>6. Преимущества и недостатки модульного подхода в проектировании.</li> <li>7. Организация крупных проектов с использованием нескольких модулей.</li> <li>8. Реализация сложных интерфейсов с использованием Tkinter или PyQt.</li> <li>9. Подключение и использование внешних библиотек для создания графического интерфейса.</li> <li>10. Реализация обработчиков событий и многозадачности в графическом интерфейсе.</li> </ol>

### Шкала оценивания ответов на вопросы

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает ответ на каждый теоретический вопрос, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает теоретические вопросы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает теоретические вопросы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не знает ответов на поставленные теоретические вопросы.

### 6.2.2. Темы для докладов

Тема (раздел)	Вопросы
Тема 1. Введение в языки программирования. Типы данных в языках	<p>УК-2</p> <ol style="list-style-type: none"> <li>1. История и эволюция языков программирования</li> <li>2. Типы данных в языке Си: основные типы</li> <li>3. Использование типов данных в базовых операциях</li> </ol>

Си.	<p>4. Работа с массивами в Си</p> <p>5. Строки и массивы символов в Си</p> <p>ОПК-6</p> <p>6. Типы данных в языке Си: указатели</p> <p>7. Массивы и структуры: как объединять данные разных типов</p> <p>8. Преобразование типов данных в Си</p> <p>9. Типы данных в Си и работа с памятью</p> <p>10. Сравнение типов данных в Си с другими языками</p>
Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции	<p>УК-2</p> <p>1. Арифметические операции в языке Си</p> <p>2. Операторы сравнения в языке Си</p> <p>3. Циклы в языке Си: for, while, do-while</p> <p>4. Условные операторы: if, switch</p> <p>5. Работа с логическими операциями в Си</p> <p>ОПК-6</p> <p>6. Рекурсия в языке Си</p> <p>7. Обработка ошибок с использованием условных операторов</p> <p>8. Циклы и их эффективность</p> <p>9. Оптимизация условий и циклов</p> <p>10. Алгоритмы с многократным использованием циклов</p>
Тема 3. Массивы. Указатели. Динамические переменные. Функции	<p>УК-2</p> <p>1. Массивы в языке Си: создание и использование</p> <p>2. Указатели: основные понятия</p> <p>3. Динамическое выделение памяти: malloc и free</p> <p>4. Работа с функциями: передача аргументов</p> <p>5. Использование указателей в функциях</p> <p>ОПК-6</p> <p>6. Массивы и указатели: как они связаны</p> <p>7. Динамические массивы: создание и управление</p> <p>8. Указатели на функции и их применение</p> <p>9. Динамическая память и управление ею</p> <p>10. Проблемы управления памятью и утечки памяти</p>
Тема 4. Сценарные языки. Язык Python.	<p>УК-2</p> <p>1. Введение в сценарные языки программирования</p> <p>2. Основы Python: синтаксис и особенности</p> <p>3. Работа с переменными и типами данных в Python</p> <p>4. Использование функций в Python</p> <p>5. Простая обработка ошибок в Python</p> <p>ОПК-6</p> <p>6. Мощь сценарных языков: Python в реальных задачах</p> <p>7. Работа с библиотеками в Python</p> <p>8. Применение Python для работы с данными</p> <p>9. Модули и пакеты в Python</p> <p>10. Сравнение Python с другими языками сценарного типа</p>
Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой	<p>УК-2</p> <p>1. Основные типы данных в Python</p> <p>2. Числа в Python: работа с целыми и вещественными числами</p> <p>3. Строки в Python: создание и методы</p>

	<ul style="list-style-type: none"> <li>4. Списки в Python: создание и манипуляция</li> <li>5. Управление логикой в Python: условные операторы</li> </ul> <p>ОПК-6</p> <ul style="list-style-type: none"> <li>6. Генераторы списков и их оптимизация</li> <li>7. Работа с кортежами и множествами</li> <li>8. Обработка строк с использованием регулярных выражений</li> <li>9. Алгоритмы сортировки списков в Python</li> <li>10. Обработка ошибок и исключений при работе с данными</li> </ul>
<p>Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.</p>	<p>УК-2</p> <ul style="list-style-type: none"> <li>1. Что такое функция в Python: синтаксис и структура</li> <li>2. Область видимости переменных: локальная и глобальная</li> <li>3. Передача данных в функцию по значению</li> <li>4. Рекурсия в Python</li> <li>5. Функции как объекты первого класса в Python</li> </ul> <p>ОПК-6</p> <ul style="list-style-type: none"> <li>6. Передача данных по ссылке в Python</li> <li>7. Замыкания и функции высшего порядка</li> <li>8. Использование анонимных функций (lambda)</li> <li>9. Декораторы в Python</li> <li>10. Типы функций и их оптимизация</li> </ul>
<p>Тема 7. Элементы функционального программирования на языке Python</p>	<p>УК-2</p> <ul style="list-style-type: none"> <li>1. Основы функционального программирования</li> <li>2. Функции как объекты первого класса в Python</li> <li>3. Применение функций высшего порядка</li> <li>4. Анонимные функции (lambda)</li> <li>5. Применение map, filter, reduce в Python</li> </ul> <p>ОПК-6</p> <ul style="list-style-type: none"> <li>6. Чистые функции и их преимущества</li> <li>7. Замыкания (closures) в Python</li> <li>8. Обработка ошибок в функциональном стиле</li> <li>9. Иммутабельность данных и её значение</li> <li>10. Использование генераторов в функциональном программировании</li> </ul>
<p>Тема 8. Модульное проектирование. Графический интерфейс на Python.</p>	<p>УК-2</p> <ul style="list-style-type: none"> <li>1. Основы модульного проектирования</li> <li>2. Создание модулей в Python</li> <li>3. Принципы инкапсуляции в Python</li> <li>4. Использование библиотеки tkinter для создания GUI</li> <li>5. Структура программы и организация модулей</li> </ul> <p>ОПК-6</p> <ul style="list-style-type: none"> <li>6. Создание и использование классов для GUI</li> <li>7. Обработка событий в графическом интерфейсе</li> <li>8. Работа с окнами и виджетами в tkinter</li> <li>9. Модульное тестирование Python-скриптов</li> <li>10. Продвинутое проектирование GUI в Python</li> </ul>

### Шкала оценивания

Шкала оценивания	Критерии оценивания
------------------	---------------------

«Отлично»	Обучающийся глубоко и содержательно раскрывает тему доклада, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает тему доклада, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает тему доклада и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой

### 6.2.3. Оценочные средства остаточных знаний (тест)

#### УК-2.

**1. Какой из этих операторов используется для условного ветвления в языке Python?**

- 1) for
- 2) if
- 3) while
- 4) else

**2. Какой результат выполнения следующего кода: `print(2**3)` в Python?**

- 1) 6
- 2) 8
- 3) 9
- 4) 16

**3. Какой тип данных в Python используется для хранения последовательности символов?**

- 1) int
- 2) str
- 3) list
- 4) tuple

**4. Как называется механизм поиска значений в списке по индексу в Python?**

- 1) Hash table
- 2) Indexing
- 3) Recursion
- 4) Binary search

**5. Какое из этих утверждений о функциях в Python неверно?**

- 1) Функции могут принимать аргументы
- 2) Функции могут возвращать значения
- 3) Функции не могут быть рекурсивными
- 4) Функции могут быть анонимными

**6. Что делает оператор continue в цикле?**

- 1) Завершает выполнение цикла
- 2) Переходит к следующей итерации цикла
- 3) Прерывает выполнение программы
- 4) Повторяет текущую итерацию

**7. Какой тип данных представляет собой неизменяемую последовательность данных в Python?**

- 1) list
- 2) tuple
- 3) dict
- 4) set

**8. Какой оператор используется для объединения строк в Python?**

- 1) +
- 2) \*
- 3) &
- 4) //

**9. Что будет результатом выполнения кода: print(10 // 3)?**

- 1) 3.33
- 2) 3
- 3) 4
- 4) 0

**10. Какой алгоритм сортировки является алгоритмом с временной сложностью  $O(n \log n)$ ?**

- 1) Пузырьковая сортировка
- 2) Сортировка слиянием
- 3) Сортировка выбором
- 4) Вставками

**11. Что такое рекурсия?**

- 1) Метод решения задачи с использованием случайных данных
- 2) Когда функция вызывает саму себя для решения подзадачи
- 3) Использование итераций в цикле
- 4) Применение метода деления и завоевания

**12. Какой алгоритм используется для поиска максимального элемента в списке?**

- 1) Линейный поиск
- 2) Бинарный поиск
- 3) Алгоритм Дейкстры
- 4) Алгоритм поиска в глубину

13. Какой из этих типов данных в Python является изменяемым?
- 1) str
  - 2) tuple
  - 3) list
  - 4) int
14. Что делает оператор **break** в цикле?
- 1) Завершает выполнение цикла и выходит из него
  - 2) Переходит к следующей итерации цикла
  - 3) Перезапускает цикл
  - 4) Повторяет текущую итерацию
15. Какую функцию нужно вызвать для получения длины строки в Python?
- 1) length()
  - 2) len()
  - 3) size()
  - 4) count()
16. Какой метод позволяет получить значение по ключу из словаря в Python?
- 1) get()
  - 2) find()
  - 3) fetch()
  - 4) search()
17. Какой из этих вариантов является правильным способом создания списка в Python?
- 1) list = [1, 2, 3]
  - 2) list = (1, 2, 3)
  - 3) list = {1, 2, 3}
  - 4) list = <1, 2, 3>
18. Какой из этих алгоритмов поиска использует отсортированный список?
- 1) Линейный поиск
  - 2) Бинарный поиск
  - 3) Поиск в глубину
  - 4) Поиск в ширину
19. Как в Python проверить, является ли элемент частью списка?
- 1) element in list
  - 2) element list
  - 3) list.has(element)
  - 4) element contains list

**20. Какое значение будет выведено при выполнении следующего кода: `print("Hello".upper())`?**

- 1) hello
- 2) Hello
- 3) HELLO
- 4) hEllo

**ОПК-6.**

**21. Какой алгоритм сортировки имеет худшую сложность  $O(n^2)$ ?**

- 1) Быстрая сортировка
- 2) Пузырьковая сортировка
- 3) Сортировка слиянием
- 4) Сортировка подсчётом

**22. Что из следующего является примером жадного алгоритма?**

- 1) Дейкстра для поиска кратчайшего пути
- 2) Алгоритм сортировки пузырьком
- 3) Алгоритм нахождения максимального подмассива
- 4) Алгоритм поиска в ширину

**23. Какой метод сортировки разделяет массив на две части и рекурсивно сортирует каждую из них?**

- 1) Сортировка слиянием
- 2) Быстрая сортировка
- 3) Пузырьковая сортировка
- 4) Сортировка вставками

**24. Как называется процесс, при котором задачи разбиваются на более мелкие подзадачи, каждая из которых решается независимо?**

- 1) Динамическое программирование
- 2) Жадный алгоритм
- 3) Разделяй и властвуй
- 4) Поиск в ширину

**25. Какой алгоритм используется для нахождения наибольшего общего делителя двух чисел?**

- 1) Алгоритм Евклида
- 2) Алгоритм Дейкстры
- 3) Алгоритм Флойда
- 4) Бинарный поиск

**26. В чем состоит суть динамического программирования?**

1) Разбиение задачи на подзадачи и сохранение решений подзадач для предотвращения повторных вычислений

2) Постепенное улучшение решения задачи на основе предыдущих результатов

3) Использование жадного подхода для решения задач

4) Использование рекурсии для решения задач

**27. Какой алгоритм используется для нахождения кратчайшего пути в графе с положительными весами?**

1) Алгоритм Дейкстры

2) Алгоритм Флойда

3) Алгоритм Беллмана-Форда

4) Поиск в ширину

**28. Какой тип сортировки является неэффективным для больших массивов данных?**

1) Быстрая сортировка

2) Пузырьковая сортировка

3) Сортировка слиянием

4) Сортировка выбором

**29. Что делает алгоритм бинарного поиска?**

1) Находит максимальный элемент в массиве

2) Ищет элемент в отсортированном массиве, деля его на части

3) Находит наибольший общий делитель двух чисел

4) Разбивает массив на два равных подмассива

**30. Как называется алгоритм, который решает задачу, всегда выбирая локально оптимальное решение на каждом шаге?**

1) Динамическое программирование

2) Жадный алгоритм

3) Алгоритм с жадным методом

4) Разделяй и властвуй

**31. Какое утверждение верно для рекурсивной функции?**

1) Она должна иметь базовый случай

2) Она всегда должна иметь несколько рекурсивных вызовов

3) Она не может использовать циклы

4) Рекурсивная функция всегда вызывает саму себя бесконечно

**32. Как называется структура данных, которая поддерживает операции добавления и удаления элементов с двух концов?**

1) Стек

2) Очередь

3) Дек

4) Множество

**33. Какой метод сортировки является неустойчивым?**

- 1) Быстрая сортировка
- 2) Сортировка слиянием
- 3) Сортировка вставками
- 4) Пузырьковая сортировка

**34. Какой алгоритм используется для поиска минимального пути во взвешенном графе?**

- 1) Алгоритм Дейкстры
- 2) Алгоритм Флойда
- 3) Алгоритм Беллмана-Форда
- 4) Алгоритм поиска в ширину

**35. Как называется метод, при котором для решения задачи используется решение меньших экземпляров той же задачи?**

- 1) Динамическое программирование
- 2) Разделение и завоевание
- 3) Рекурсия
- 4) Жадный алгоритм

**36. Какой из этих алгоритмов является примером сортировки с временной сложностью  $O(n \log n)$ ?**

- 1) Быстрая сортировка
- 2) Пузырьковая сортировка
- 3) Сортировка выбором
- 4) Сортировка вставками

**37. Какой из этих алгоритмов является методом слияния отсортированных массивов?**

- 1) Сортировка слиянием
- 2) Быстрая сортировка
- 3) Пузырьковая сортировка
- 4) Сортировка вставками

**38. Что такое "хеш-функция"?**

- 1) Функция для поиска элемента в массиве
- 2) Функция, которая преобразует данные произвольного размера в фиксированное значение
- 3) Функция для шифрования данных
- 4) Функция для сортировки данных

**39. Как называется структура данных, которая хранит элементы в порядке их добавления?**

- 1) Стек
- 2) Очередь

- 3) Дек
- 4) Множество

**40. Что из этого является примером рекурсивного алгоритма?**

- 1) Алгоритм поиска в глубину
- 2) Алгоритм сортировки пузырьком
- 3) Алгоритм сортировки вставками
- 4) Алгоритм быстрой сортировки

**41. Какое время работы алгоритма сортировки слиянием?**

- 1)  $O(n)$
- 2)  $O(n^2)$
- 3)  $O(n \log n)$
- 4)  $O(\log n)$

**42. Какой тип алгоритма поиска подходит для поиска элемента в отсортированном массиве?**

- 1) Линейный поиск
- 2) Бинарный поиск
- 3) Поиск в глубину
- 4) Поиск в ширину

**43. Какой из этих алгоритмов является жадным методом?**

- 1) Алгоритм Дейкстры
- 2) Быстрая сортировка
- 3) Пузырьковая сортировка
- 4) Алгоритм Флойда

**44. Какой метод сортировки использует стратегию "разделяй и властвуй"?**

- 1) Быстрая сортировка
- 2) Пузырьковая сортировка
- 3) Сортировка вставками
- 4) Сортировка выбором

**45. Какой из следующих алгоритмов является методом с использованием динамического программирования?**

- 1) Алгоритм нахождения наибольшей общей подпоследовательности
- 2) Алгоритм нахождения кратчайшего пути
- 3) Алгоритм поиска в глубину
- 4) Алгоритм быстрой сортировки

**Ключ к тесту:**

1.2	2.2	3.2	4.2	5.3	6.2	7.2	8.1	9.2
-----	-----	-----	-----	-----	-----	-----	-----	-----

<b>10.2</b>	<b>11.2</b>	<b>12.1</b>	<b>13.3</b>	<b>14.1</b>	<b>15.2</b>	<b>16.1</b>	<b>17.1</b>	<b>18.2</b>
<b>19.1</b>	<b>20.3</b>	<b>21.2</b>	<b>22.1</b>	<b>23.1</b>	<b>24.3</b>	<b>25.1</b>	<b>26.1</b>	<b>27.1</b>
<b>28.2</b>	<b>29.2</b>	<b>30.2</b>	<b>31.1</b>	<b>32.3</b>	<b>33.1</b>	<b>34.1</b>	<b>35.3</b>	<b>36.1</b>
<b>37.1</b>	<b>38.2</b>	<b>39.2</b>	<b>40.1</b>	<b>41.3</b>	<b>42.2</b>	<b>43.1</b>	<b>44.1</b>	<b>45.1</b>

### **Шкала оценивания результатов тестирования**

% верных решений (ответов)	Шкала оценивания
85 - 100	отлично
70 - 84	хорошо
50- 69	удовлетворительно
0 - 49	неудовлетворительно

#### **6.2.4. Примеры задач при разборе конкретных ситуаций**

*Тема 1. Введение в языки программирования. Типы данных в языках Си.*

##### **УК-2.**

1. Написать обзор на различные языки программирования, акцентируя внимание на языке Си, его особенностях и областях применения.

2. Описать основные типы данных в языке Си (int, float, char и т.д.) и их применение в программировании. Привести примеры кода, демонстрирующие использование различных типов данных.

##### **ОПК-6.**

3. Разработать программу на языке Си, которая принимает ввод от пользователя и выводит тип данных для каждого введенного значения.

4. Создать таблицу, в которой будут перечислены различные типы данных в языке Си, их размер в памяти и примеры использования.

*Тема 2. Основные операции в языке Си. Базовые алгоритмические конструкции*

##### **УК-2.**

1. Описать основные операции в языке Си: арифметические, логические и побитовые операции. Привести примеры их использования в коде.

2. Рассмотреть базовые алгоритмические конструкции, такие как условные операторы и циклы. Написать примеры программ, использующих эти конструкции.

##### **ОПК-6.**

3. Написать программу на языке Си, которая использует циклы для вычисления факториала числа, введенного пользователем.

4. Создать программу, которая демонстрирует использование условных операторов для определения, является ли число четным или нечетным.

*Тема 3. Массивы. Указатели. Динамические переменные. Функции*

##### **УК-2.**

1. Описать, что такое массивы в языке Си, и привести примеры их использования.

2. Объяснить, как работают указатели в языке Си, и написать программу, которая использует указатели для обмена значениями двух переменных.

**ОПК-6.**

3. Разработать программу на языке Си, которая динамически выделяет память для массива целых чисел и заполняет его значениями, введенными пользователем.

4. Написать функцию, которая принимает массив и его размер, и возвращает сумму всех элементов массива.

*Тема 4. Сценарные языки. Язык Python.*

**УК-2.**

1. Написать обзор на сценарные языки программирования, акцентируя внимание на языке Python и его особенностях.

2. Описать основные конструкции языка Python, такие как условные операторы, циклы и функции, с примерами кода.

**ОПК-6.**

3. Создать простую программу на Python, которая запрашивает у пользователя имя и приветствует его, используя введенное имя.

4. Написать программу на Python, которая выполняет простые математические операции (сложение, вычитание, умножение, деление) на основе пользовательского ввода.

*Тема 5. Базовые типы данных. Числа. Строки. Списки. Средства управления логикой*

**УК-2.**

1. Описать базовые типы данных в Python: числа, строки и списки. Привести примеры кода, демонстрирующие их использование.

2. Рассмотреть средства управления логикой в Python, такие как условные операторы и логические выражения. Написать примеры кода, использующие эти конструкции.

**ОПК-6.**

3. Разработать программу на Python, которая принимает ввод чисел от пользователя и выводит их сумму и среднее значение.

4. Написать программу, которая обрабатывает строку (например, считает количество слов и символов) и выводит результаты на экран.

*Тема 6. Функции. Понятие области видимости переменных. Передача данных в функцию.*

**УК-2.**

1. Описать концепцию функций в Python, включая параметры и возвращаемые значения. Привести примеры функций с различными параметрами.

2. Объяснить понятие области видимости переменных и как она работает в Python.

**ОПК-6.**

3. Написать программу на Python, которая определяет функцию для вычисления факториала и использует её для вычисления факториала числа, введенного пользователем.

4. Создать программу, которая демонстрирует передачу данных в функцию, включая использование аргументов по умолчанию и переменное количество аргументов.

*Тема 7. Элементы функционального программирования на языке Python*

#### **УК-2.**

1. Описать основные элементы функционального программирования и как они реализованы в Python, включая функции высшего порядка и лямбда-функции.

2. Рассмотреть использование встроенных функций Python, таких как `map()`, `filter()` и `reduce()`.

#### **ОПК-6.**

3. Написать программу на Python, которая использует лямбда-функции для сортировки списка словарей по заданному ключу.

4. Создать программу, которая использует функции `map()` и `filter()` для обработки списка чисел, выводя только четные числа и их квадрат.

*Тема 8. Модульное проектирование. Графический интерфейс на Python.*

#### **УК-2.**

1. Описать принципы модульного проектирования и как они применяются в Python, включая использование модулей и пакетов.

2. Рассмотреть основные библиотеки для создания графического интерфейса на Python, такие как Tkinter.

#### **ОПК-6.**

3. Разработать простое графическое приложение на Python с использованием Tkinter, которое позволяет пользователю вводить текст и выводить его на экран.

4. Написать программу, которая создает графический интерфейс для калькулятора, позволяя пользователю выполнять базовые арифметические операции.

### **Шкала оценивания**

Шкала оценивания	Критерии оценивания
«Отлично»	обучающийся ясно изложил условие задачи, решение обосновал
«Хорошо»	обучающийся ясно изложил условие задачи, но в обосновании решения имеются сомнения;
«Удовлетворительно»	обучающийся изложил решение задачи, но обосновал его формулировками обыденного мышления;
«Неудовлетворительно»	обучающийся не уяснил условие задачи, решение не обосновал либо не сдал работу на проверку (в случае проведения решения задач в письменной форме).

#### **6.2.5. Индивидуальные задания для курсовой работы (проекта)**

Обучающимся предоставляется право выбора темы курсовой работы в соответствии с разработанным перечнем, или обучающийся может предложить свою тему с обоснованием ее актуальности и целесообразности исследования.

Во всех случаях тема курсовой работы должна быть согласована с научным руководителем.

Методические указания к курсовой работе по дисциплине «Программирование и основы алгоритмизации» к рабочей программе дисциплины прилагаются.

### **Тематика курсовых работ**

1. Приложение для учета работы школы робототехники
2. Приложение для учета работы компьютерного клуба
3. Приложение для учета работы серверной
4. Приложение для учета работы сетевого оборудования
5. Приложение для учета работы вычислительного центра
6. Приложение для учета работы отдела информационных технологий
7. Приложение для учета работы лаборатории микропроцессоров
8. Приложение для учета работы студии разработки программ
9. Приложение для учета работы центра обработки данных
10. Приложение для учета работы учебного класса информатики
11. Приложение для учета работы мастерской по ремонту компьютеров
12. Приложение для учета работы отдела разработки
13. Приложение для учета работы системного администратора
14. Приложение для учета работы центра цифрового образования
15. Приложение для учета работы лаборатории умных устройств
16. Приложение для учета работы службы компьютерной безопасности
17. Приложение для учета работы пункта проката техники
18. Приложение для учета работы сервисного центра
19. Приложение для учета работы факультета информатики
20. Приложение для учета работы научной лаборатории
21. Приложение для учета работы отдела тестирования программ
22. Приложение для учета работы технической поддержки
23. Приложение для учета работы центра хранения данных
24. Приложение для учета работы полигона системного администрирования
25. Приложение для учета работы отдела защиты информации
26. Приложение для учета работы студии создания приложений
27. Приложение для учета работы центра дистанционного обучения
28. Приложение для учета работы проектного офиса
29. Приложение для учета работы лаборатории искусственного интеллекта
30. Приложение для учета работы класса программирования
31. Приложение для учета работы компьютерного центра
32. Приложение для учета работы отдела автоматизации
33. Приложение для учета работы группы внедрения программ
34. Приложение для учета работы центра пространственных данных
35. Приложение для учета работы лаборатории робототехники

36. Приложение для учета работы школы алгоритмики
37. Приложение для учета работы центра компьютерного спорта
38. Приложение для учета работы отдела контроля качества программ
39. Приложение для учета работы вычислительной платформы
40. Приложение для учета работы лаборатории обработки сигналов
41. Приложение для учета работы группы разработки системных программ
42. Приложение для учета работы сборочного цеха компьютеров
43. Приложение для учета работы центра проверки безопасности
44. Приложение для учета работы класса операционных систем
45. Приложение для учета работы лаборатории высокопроизводительных вычислений
46. Приложение для учета работы отдела администрирования
47. Приложение для учета работы группы разработки хранилищ данных
48. Приложение для учета работы центра создания образцов
49. Приложение для учета работы школы защиты информации
50. Приложение для учета работы лаборатории связанных устройств
51. Приложение для учета работы отдела обучающих систем
52. Приложение для учета работы центра отечественного программного обеспечения
53. Приложение для учета работы группы технического надзора
54. Приложение для учета работы студии виртуальной реальности
55. Приложение для учета работы лаборатории компьютерной графики
56. Приложение для учета работы отдела цифрового управления
57. Приложение для учета работы центра управления сетями
58. Приложение для учета работы школы системного анализа
59. Приложение для учета работы мастерской по ремонту ноутбуков
60. Приложение для учета работы лаборатории шифрования
61. Приложение для учета работы центра обработки текстов
62. Приложение для учета работы отдела сопровождения систем
63. Приложение для учета работы группы автоматизации проверок
64. Приложение для учета работы центра сохранения копий
65. Приложение для учета работы школы создания игр
66. Приложение для учета работы лаборатории биологических данных
67. Приложение для учета работы отдела систем контроля доступа
68. Приложение для учета работы центра компетенций по операционным системам
69. Приложение для учета работы группы наблюдения за системами
70. Приложение для учета работы лаборатории измерительных устройств
71. Приложение для учета работы школы работы с данными
72. Приложение для учета работы отдела создания интерфейсов
73. Приложение для учета работы центра защиты личных данных
74. Приложение для учета работы группы переноса систем
75. Приложение для учета работы лаборатории встраиваемых систем
76. Приложение для учета работы центра управления работами

77. Приложение для учета работы школы разработки и сопровождения  
 78. Приложение для учета работы отдела соединения систем  
 79. Приложение для учета работы центра цифровых умений  
 80. Приложение для учета работы группы поддержки веб-служб

### Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	<p>ставится за курсовую работу, которая характеризуется использованием большого количества новейших литературных источников, глубоким анализом привлеченного материала, творческим подходом к его изложению, знанием закономерностей функционирования современных информационных систем, основных понятий, категорий и инструментов в области информатики и вычислительной техники, основных особенностей ведущих школ и направлений в сфере ИТ; умением анализировать и интерпретировать данные отечественной и зарубежной статистики об информационных процессах и явлениях, выявлять тенденции, прогнозировать возможность их развития в будущем, выявлять проблемы технического и алгоритмического характера при анализе конкретных ситуаций, предлагать способы их решения, оценивать риски и возможные технические последствия тех или иных явлений, происходящих в сфере информатики и вычислительной техники. Работа по НИР получает наивысшую оценку в случае одновременного выполнения следующих условий:</p> <p style="padding-left: 40px;">а) объект исследования описан с предельно широким привлечением источников (как внутренних, так и внешних), на него составлено соответствующее досье, в которое скопированы все использованные материалы;</p> <p style="padding-left: 40px;">б) самостоятельно и корректно (т.е. в соответствии с реальными фактами) сделаны выводы из анализа досье;</p> <p style="padding-left: 40px;">в) выявлена взаимосвязь полученных результатов с общетеоретическими проблемами курса микроэкономики.</p> <p>Вынесенные в Приложение материалы могут повысить общую оценку за курсовую работу.</p>
«Хорошо»	<p>ставится за курсовую работу, написанную на достаточно высоком теоретическом уровне, в полной мере раскрывающую содержание темы курсовой работы, с приведенным фактическим материалом, по которому сделаны правильные выводы и обобщения, произведена увязка теории с практикой современной действительности, правильно оформленную работу.</p>
«Удовлетворительно»	<p>ставится за курсовую работу, в которой недостаточно полно освещены узловые вопросы темы, работа написана на базе очень небольшого количества источников, либо на базе устаревших источников.</p>
«Неудовлетворительно»	<p>ставится за курсовую работу, переписанную с одного или нескольких источников. Работа в рамках НИР оценивается неудовлетворительно в случае нарушения требований задания.</p>

## 6.3. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОГО КОНТРОЛЯ

**Вопросы для подготовки к промежуточной аттестации по итогам освоения дисциплины Программирование и основы алгоритмизации:**

**УК-2.**

1. Введение в языки программирования

2. Понятие программа
3. Интерпретация и компиляция
4. Понятие шаблонный код
5. Понятие тип данных, операция
6. Базовые типы данных в языках Си
7. Взаимодействие программы с пользователем
8. Библиотека `stdio.h`
9. Понятие переменная
10. Операция присваивания
11. Динамическая и статическая типизация
12. Основные операции в языке Си
13. Приоритеты операций
14. Вычисление выражений
15. Явное и неявное приведение типов
16. Типовые ошибки при вычислении выражений в языке Си
17. Базовые алгоритмические конструкции
18. Ветвление
19. Виды ветвлений
20. Блок кода
21. Понятие области видимости переменных
22. Цикл `while`, `for`
23. Выход из цикла `break`, `continue`
24. Массивы
25. Определение размера массива
26. Инициализация массива
27. Генерация случайных чисел пользователями
28. Понятие интерфейс
29. Указатели. Ссылки. Операции с указателями
30. Способы инициализации массивов
31. Динамические переменные
32. Управление памятью в Си
33. Динамические массивы
34. Реализация массивов переменной длины
35. Строки в Си как массив символов
36. Операции со строками
37. Составные типы данных на языке Си
38. Структуры, перечисления, объединения
39. Указатели на структуру
40. Получение данных из командной строки
41. Понятие чистой функции
42. Прототип функции
43. Рекурсия
44. Функции итераторы и генераторы
45. Указатели на функцию
46. Функции первого порядка

47. Элементы функционального программирования на языке Си
48. Алгоритмы сортировки: пузырьковая, выбором, вставками.
49. Алгоритм поиска элемента в массиве: линейный и бинарный поиск.
50. Принцип «разделяй и властвуй» и его применение.
51. Использование стека и очереди в алгоритмах.
52. Алгоритмы обработки строк: подсчёт символов, поиск подстроки.
53. Построение таблиц частот символов.
54. Использование двумерных массивов и алгоритмы над ними.
55. Построение и обход графов (вводное понятие).

#### **ОПК-6.**

56. Обзор современных языков программирования
57. Среды разработки
58. Реализация типов данных
59. Способы взаимодействия программы с пользователем
60. Файловый ввод – вывод
61. Работа с файлами на языке Си
62. Типы файлов
63. Функции. Передача данных в функцию по значению и ссылке
64. Проектирование приложений с помощью функций
65. Модификаторы области видимости переменных
66. Структура типового проекта
67. Заголовочные файлы
68. Библиотеки функций
69. Этапы разработки программного обеспечения: анализ, проектирование, реализация, тестирование.
70. Методология разработки ПО: водопадная модель, Agile.
71. Понятие отладки и типичные ошибки времени выполнения.
72. Инструменты для профилирования и отладки программ.
73. Принципы модульного программирования.
74. Понятие инкапсуляции и модульной абстракции.
75. Принципы тестирования программ: юнит-тестирование, тесты на граничные значения.

#### **6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Основной целью проведения промежуточной аттестации является определение степени достижения целей по учебной дисциплине или ее разделам. Осуществляется это проверкой и оценкой уровня теоретической знаний, полученных обучающимися, умения применять их в решении практических задач, степени овладения обучающимися практическими

навыками и умениями в объеме требований рабочей программы по дисциплине, а также их умение самостоятельно работать с учебной литературой.

Организация проведения промежуточной аттестации регламентирована «Положением об организации образовательного процесса в федеральном государственном автономном образовательном учреждении «Московский политехнический университет»

#### 6.4.1. Показатели оценивания компетенций на различных этапах их формирования, достижение обучающимися планируемых результатов обучения по дисциплине

Код и наименование компетенции УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений				
Этап (уровень)	Критерии оценивания			
	неудовлетворительно	удовлетворительно	хорошо	отлично
<b>знать</b>	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность направления подготовки	Обучающийся демонстрирует неполное соответствие следующих знаний: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность направления подготовки	Обучающийся демонстрирует частичное соответствие следующих знаний: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность направления подготовки	Обучающийся демонстрирует полное соответствие следующих знаний: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность направления подготовки
<b>уметь</b>	Обучающийся не умеет или в недостаточной степени умеет: проводить анализ поставленной цели, формулировать проблему, решение которой связано с достижением цели проекта и задачи, которые необходимо решить для ее достижения;	Обучающийся демонстрирует неполное соответствие следующих умений: проводить анализ поставленной цели, формулировать проблему, решение которой связано с достижением цели проекта и задачи, которые необходимо решить для ее	Обучающийся демонстрирует частичное соответствие следующих умений: проводить анализ поставленной цели, формулировать проблему, решение которой связано с достижением цели проекта и задачи, которые	Обучающийся демонстрирует полное соответствие следующих умений: проводить анализ поставленной цели, формулировать проблему, решение которой связано с

	анализировать альтернативные варианты для достижения намеченных результатов и выбирать оптимальные способы их решения; использовать нормативно-правовую документацию в сфере профессиональной деятельности направления подготовки	достижения; анализировать альтернативные варианты для достижения намеченных результатов и выбирать оптимальные способы их решения; использовать нормативно-правовую документацию в сфере профессиональной деятельности направления подготовки	необходимо решить для ее достижения; анализировать альтернативные варианты для достижения намеченных результатов и выбирать оптимальные способы их решения; использовать нормативно-правовую документацию в сфере профессиональной деятельности направления подготовки	достижением цели проекта и задачи, которые необходимо решить для ее достижения; анализировать альтернативные варианты для достижения намеченных результатов и выбирать оптимальные способы их решения; использовать нормативно-правовую документацию в сфере профессиональной деятельности направления подготовки
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>владеть</b>	Обучающийся не владеет или в недостаточной степени владеет навыками постановки цели и задач проекта; методиками оценки потребности в ресурсах, продолжительности и стоимости проекта; навыками работы с нормативно-правовой документацией	Обучающийся владеет в неполном объеме и проявляет недостаточность владения навыками постановки цели и задач проекта; методиками оценки потребности в ресурсах, продолжительности и стоимости проекта; навыками работы с нормативно-правовой документацией	Обучающимся допускаются незначительные ошибки, неточности, затруднения, частично владеет навыками постановки цели и задач проекта; методиками оценки потребности в ресурсах, продолжительности и стоимости проекта; навыками работы с нормативно-правовой документацией	Обучающийся свободно применяет полученные навыки, в полном объеме владеет навыками постановки цели и задач проекта; методиками оценки потребности в ресурсах, продолжительности и стоимости проекта; навыками работы с нормативно-правовой документацией
----------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Код и наименование компетенции ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием**

Этап (уровень)	Критерии оценивания			
	неудовлетворительно	удовлетворительно	хорошо	отлично
<b>знать</b>	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: методы	Обучающийся демонстрирует неполное соответствие следующих знаний: методы алгоритмизации, языки	Обучающийся демонстрирует частичное соответствие следующих знаний: методы	Обучающийся демонстрирует полное соответствие следующих знаний: методы

	алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий	и технологии программирования, пригодные для практического применения в области информационных систем и технологий	алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий	алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий
<b>уметь</b>	Обучающийся не умеет или в недостаточной степени умеет: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий	Обучающийся демонстрирует неполное соответствие следующих умений: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий	Обучающийся демонстрирует частичное соответствие следующих умений: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий	Обучающийся демонстрирует полное соответствие следующих умений: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий
<b>владеть</b>	Обучающийся не владеет или в недостаточной степени владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач	Обучающийся владеет в неполном объеме и проявляет недостаточность владения навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач	Обучающимся допускаются незначительные ошибки, неточности, затруднения, частично владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач	Обучающийся свободно применяет полученные навыки, в полном объеме владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

#### 6.4.2. Методика оценивания результатов промежуточной аттестации

Показателями оценивания компетенций на этапе промежуточной аттестации по дисциплине «Программирование и основы алгоритмизации» являются результаты обучения по дисциплине.

#### Оценочный лист результатов обучения по дисциплине

Код компетенции	Знания	Умения	Навыки	Уровень сформированности компетенции на данном этапе / оценка
-----------------	--------	--------	--------	---------------------------------------------------------------

УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	на уровне знаний: знать методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.	на уровне умений: уметь применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий.	на уровне навыков: программирования, отладки и тестирования прототипов программно-технических комплексов задач.	
ОПК-6. Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	на уровне знаний: знать методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.	на уровне умений: уметь применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий.	на уровне навыков: программирования, отладки и тестирования прототипов программно-технических комплексов задач..	
Оценка по дисциплине (среднее арифметическое)				

Оценка «отлично» выставляется, если среднее арифметическое находится в интервале от 4,5 до 5,0.

Оценка «хорошо» выставляется, если среднее арифметическое находится в интервале от 3,5 до 4,4.

Оценка «удовлетворительно» выставляется, если среднее арифметическое находится в интервале от 2,5 до 3,4.

Оценка «неудовлетворительно» выставляется, если среднее арифметическое находится в интервале от 0 до 2,4.

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по дисциплине «Программирование и основы алгоритмизации», при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися

планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

<b>Шкала оценивания</b>	<b>Описание</b>
Зачтено	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Не зачтено	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков по этапам (уровням) сформированности компетенций, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

<b>Шкала оценивания</b>	<b>Описание</b>
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду

	показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
--	----------------------------------------------------------------------------------------------------------------------------------

## **7. Электронная информационно-образовательная среда**

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронной информационно-образовательной среде Чебоксарского института (филиала) Московского политехнического университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), как на территории филиала, так и вне ее.

Электронная информационно-образовательная среда – совокупность информационных и телекоммуникационных технологий, соответствующих технологических средств, обеспечивающих освоение обучающимися образовательных программ в полном объеме независимо от места нахождения обучающихся.

Электронная информационно-образовательная среда обеспечивает:

а) доступ к учебным планам, рабочим программам дисциплин (модулей), практик, электронным учебным изданиям и электронным образовательным ресурсам, указанным в рабочих программах дисциплин (модулей), практик;

б) формирование электронного портфолио обучающегося, в том числе сохранение его работ и оценок за эти работы;

в) фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения программы бакалавриата;

г) проведение учебных занятий, процедур оценки результатов обучения, реализация которых предусмотрена с применением электронного обучения, дистанционных образовательных технологий;

д) взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействия посредством сети «Интернет».

Функционирование электронной информационно-образовательной среды обеспечивается соответствующими средствами информационно-коммуникационных технологий и квалификацией работников, ее использующих и поддерживающих.

Функционирование электронной информационно-образовательной среды соответствует законодательству Российской Федерации.

Основными составляющими ЭИОС филиала являются:

а) сайт института в сети Интернет, расположенный по адресу [www.polytech21.ru](http://www.polytech21.ru), <https://chebpolytech.ru/> который обеспечивает:

- доступ обучающихся к учебным планам, рабочим программам дисциплин, практик, к изданиям электронных библиотечных систем, электронным информационным и образовательным ресурсам, указанных в рабочих программах (разделы сайта «Сведения об образовательной организации»);

- информирование обучающихся обо всех изменениях учебного процесса (новостная лента сайта, лента анонсов);

- взаимодействие между участниками образовательного процесса (подразделы сайта «Задать вопрос директору»);

б) официальные электронные адреса подразделений и сотрудников института с Яндекс-доменом @polytech21.ru (список контактных данных подразделений Филиала размещен на официальном сайте Филиала в разделе «Контакты», списки контактных официальных электронных данных преподавателей размещены в подразделах «Кафедры») обеспечивают взаимодействие между участниками образовательного процесса;

в) личный кабинет обучающегося (портфолио) (вход в личный кабинет размещен на официальном сайте Филиала в разделе «Студенту» подразделе «Электронная информационно-образовательная среда») включает в себя портфолио студента, электронные ведомости, рейтинг студентов и обеспечивает:

- фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися,

- формирование электронного портфолио обучающегося, в том числе с сохранение работ обучающегося, рецензий и оценок на эти работы,

г) электронные библиотеки, включающие электронные каталоги, полнотекстовые документы и обеспечивающие доступ к учебно-методическим материалам, выпускным квалификационным работам и т.д.:

Чебоксарского института (филиала) - «ИРБИС»

д) электронно-библиотечные системы (ЭБС), включающие электронный каталог и полнотекстовые документы:

- ЭБС «ЛАНЬ» -<https://e.lanbook.com/>

- Образовательная платформа Юрайт - <https://urait.ru>

- IPR SMART -<https://www.iprbookshop.ru/>

е) платформа цифрового образования Политеха - <https://lms.mospolytech.ru/>

ж) система «Антиплагиат» -<https://www.antiplagiat.ru/>

з) система электронного документооборота DIRECTUM Standard — обеспечивает документооборот между Филиалом и Университетом;

и) система «1С Управление ВУЗом Электронный деканат» (Московский политехнический университет) обеспечивает фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися;

к) система «POLYTECH systems» обеспечивает информационное, документальное автоматизированное сопровождение образовательного процесса;

л) система «Абитуриент» обеспечивает документальное автоматизированное сопровождение работы приемной комиссии.

## **8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

#### Основная литература:

1. Огнева, М. В. Программирование на языке C++: практический курс : учебник для вузов / М. В. Огнева, Е. В. Кудрина, А. А. Казачкова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 342 с. — (Высшее образование). — ISBN 978-5-534-18949-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/563618>.
2. Федоров, Д. Ю. Программирование на python : учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 187 с. — (Высшее образование). — ISBN 978-5-534-19666-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/556864>.
3. Трофимов, В. В. Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Павловская. — 4-е изд. — Москва : Издательство Юрайт, 2025. — 108 с. — (Высшее образование). — ISBN 978-5-534-20430-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562040>.
4. Кудрина, Е. В. Основы алгоритмизации и программирования на языке C# : учебник для вузов / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2025. — 322 с. — (Высшее образование). — ISBN 978-5-534-09796-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/565466>.

#### Дополнительная литература:

1. Гниденко, И. Г. Технологии и методы программирования : учебник для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 241 с. — (Высшее образование). — ISBN 978-5-534-18130-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/581329>.
2. Кувшинов, Д. Р. Программирование на c : учебное пособие для вузов / Д. Р. Кувшинов. — Москва : Издательство Юрайт, 2025. — 83 с. — (Высшее образование). — ISBN 978-5-534-21174-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/559503>.
3. Якимов, С. П. Структурное программирование : учебник для вузов / С. П. Якимов. — Москва : Издательство Юрайт, 2025. — 342 с. — (Высшее образование). — ISBN 978-5-534-14885-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/567948>.

#### Периодика:

1. Известия Тульского государственного университета. Технические науки: Научный рецензируемый журнал. <https://tidings.tsu.tula.ru/tidings/index.php?id=technical&lang=ru&year=1>.  
- Текст : электронный.
2. Научный периодический журнал «Вестник Южно-Уральского государственного университета. Серия «Вычислительная математика и

информатика» : Научный рецензируемый журнал. <https://vestnik.susu.ru/cmi> - Текст : электронный.

3. Научный периодический журнал «Вестник Южно-Уральского государственного университета. Серия «Компьютерные технологии, управление, радиоэлектроника»: Научный рецензируемый журнал. <https://vestnik.susu.ru/ctcr> - Текст : электронный.

## 9. Профессиональные базы данных и информационно-справочные системы

Профессиональная база данных и информационно-справочные системы	Информация о праве собственности (реквизиты договора)
Университетская информационная система РОССИЯ <a href="https://uisrussia.msu.ru/">https://uisrussia.msu.ru/</a>	Тематическая электронная библиотека и база для прикладных исследований в области экономики, управления, социологии, лингвистики, философии, филологии, международных отношений, права. свободный доступ
научная электронная библиотека Elibrary <a href="http://elibrary.ru/">http://elibrary.ru/</a>	Научная электронная библиотека eLIBRARY.RU - это крупнейший российский информационно-аналитический портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты более 26 млн научных статей и публикаций, в том числе электронные версии более 5600 российских научно-технических журналов, из которых более 4800 журналов в открытом доступе свободный доступ
сайт Института научной информации по общественным наукам РАН. <a href="http://www.inion.ru">http://www.inion.ru</a>	Библиографические базы данных ИНИОН РАН по социальным и гуманитарным наукам ведутся с начала 1980-х годов. Общий объём массивов составляет более 3 млн. 500 тыс. записей (данные на 1 января 2012 г.). Ежегодный прирост — около 100 тыс. записей. В базы данных включаются аннотированные описания книг и статей из журналов и сборников на 140 языках, поступивших в Фундаментальную библиотеку ИНИОН РАН. Описания статей и книг в базах данных снабжены шифром хранения и ссылками на полные тексты источников из Научной электронной библиотеки.
Федеральный портал «Российское образование» [Электронный ресурс] – <a href="http://www.edu.ru">http://www.edu.ru</a>	Федеральный портал «Российское образование» – уникальный интернет-ресурс в сфере образования и науки. Ежедневно публикует самые актуальные новости, анонсы событий, информационные материалы для широкого круга читателей. Еженедельно на портале размещаются эксклюзивные материалы, интервью с ведущими специалистами – педагогами, психологами, учеными, репортажи и аналитические статьи. Читатели получают доступ к нормативно-правовой базе сферы образования, они могут пользоваться самыми различными полезными сервисами – такими, как онлайн-тестирование, опросы по актуальным темам и т.д.
Информационные технологии – периодическое научно-техническое издание	Издательство выпускает теоретические и прикладные научно-технические журналы, обеспечивающие научной, производственной, обзорно-аналитической и

в области информационных технологий, автоматизированных систем и использования информатики в различных приложениях novtex.ru	образовательной информацией руководящих работников и специалистов промышленных предприятий, научных академических и отраслевых организаций, а также учебных заведений в области приоритетных направлений развития науки и технологий.
Ассоциация инженерного образования России <a href="http://www.ac-raee.ru/">http://www.ac-raee.ru/</a>	Совершенствование образования и инженерной деятельности во всех их проявлениях, относящихся к учебному, научному и технологическому направлениям, включая процессы преподавания, консультирования, исследования, разработки инженерных решений, оказания широкого спектра образовательных услуг, обеспечения связей с общественностью, производством, наукой и интеграции в международное научно-образовательное пространство. свободный доступ

### 10. Программное обеспечение (лицензионное и свободно распространяемое), используемое при осуществлении образовательного процесса

Аудитория	Программное обеспечение	Информация о праве собственности (реквизиты договора, номер лицензии и т.д.)
№ 2196 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП»	Windows 7 OLPNLAcdmc	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025
	Yandex браузер	Свободное распространяемое программное обеспечение (бессрочная лицензия)
	МТС Линк	Договор №2/2026 (091_168.ЕП/26) от 27.03.2026
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic (Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
№ 2066 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025
	Windows 7 OLPNLAcdmc Windows Server 2012	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	MS Windows 10 Pro	договор № 392_469.223.3К/19 от

<p>магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей)</p> <p><u>Лаборатория «Программирования и баз данных»</u></p> <p><u>Лаборатория информационных технологий</u></p>	Microsoft Office Standard 2019(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License	17.12.19 (бессрочная лицензия) номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Eclipse IDE for Java EE Developers, NET Framework, JDK 8, Microsoft SQL Server Express Edition, Microsoft Visio Professional, Microsoft Visual Studio, MySQL Installer for Windows, NetBeans, SQL Server Management Studio, Microsoft SQL Server Java Connector, Android Studio, IntelliJ IDEA.	свободно распространяемое программное обеспечение (бессрочная лицензия)
	КОМПАС-3D v20 и v21	Сублицензионный договор № Нп-22-00044 от 21.03.2022 (бессрочная лицензия)
	MathCADv.15	Сублиц.договор №39331/МОС2286 от 6.05.2013) номер лицензии-42661846 от 30.08.2007) (бессрочная лицензия)
	SimInTech	Отечественное программное обеспечение
	AdobeReader	свободно распространяемое программное обеспечение (бессрочная лицензия)
	AdobeFlashPlayer	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Visual Studio 2019	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Python 3.7	свободно распространяемое программное обеспечение (бессрочная лицензия)
	PascalABC	свободно распространяемое программное обеспечение (бессрочная лицензия)
	AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)
<p><b>№ 1126</b> Помещение для самостоятельной работы обучающихся</p>	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025
	Windows 7 OLPNLAcdmс	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	AdobeReader	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Гарант- справочно-правовая система	Договор №С-002-2025 от 09.01.2025

	Yandex браузер	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic (Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	МТС Линк	Договор №2/2026 (091_168.ЕП/26) от 27.03.2026
	AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)

## 11. Материально-техническое обеспечение дисциплины

Тип и номер помещения	Перечень основного оборудования и технических средств обучения
Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/ специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП» № 2196 (г. Чебоксары, ул. К.Маркса, 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; стенды <u>Технические средства обучения:</u> мультимедийное оборудование (проектор, экран)
Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/ специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Лаборатория информационных технологий № 2066 (г. Чебоксары, ул. К.Маркса, 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; автоматизированные рабочие места, автоматизированное рабочее место преподавателя, проектор и экран; маркерная доска; <u>Технические средства обучения:</u> компьютерная техника (процессор Core i3, оперативная память объемом не менее 8 Гб); сервер в лаборатории (8-ядерный процессор с частотой не менее 3 ГГц, оперативная память объемом не менее 16 Гб, жесткие диски общим объемом не менее 1 Тб; мультимедийное оборудование (телевизор)
Помещение для самостоятельной работы обучающихся № 1126 (г. Чебоксары, ул. К.Маркса, 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; <u>Технические средства обучения:</u> компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Филиала

## 12. Методические указания для обучающегося по освоению дисциплины

### *Методические указания для занятий лекционного типа*

В ходе лекционных занятий обучающемуся необходимо вести конспектирование учебного материала, обращать внимание на категории,

формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации.

Необходимо задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций. Целесообразно дорабатывать свой конспект лекции, делая в нем соответствующие записи из основной и дополнительной литературы, рекомендованной преподавателем и предусмотренной учебной программой дисциплины.

#### ***Методические указания для занятий лабораторного типа.***

Выполнению лабораторных работ предшествует проверка знаний студентов – их теоретической готовности к выполнению задания. Проверка знаний проводится в форме, которую определяет преподаватель дисциплины (тестирование, опрос).

При проведении лабораторных занятий выделяют следующие разделы:

- общие положения (перечень лабораторных или практических занятий);
- общие требования к выполнению работ;
- инструкция по каждой работе;
- справочные материалы и т. д.

Лабораторные занятия позволяют развивать у обучающегося творческое теоретическое мышление, умение самостоятельно изучать литературу, анализировать практику; учат четко формулировать мысль, вести дискуссию, то есть имеют исключительно важное значение в развитии самостоятельного мышления.

Готовясь к докладу или выступлению в рамках интерактивной формы, при необходимости, следует обратиться за помощью к преподавателю.

#### ***Методические указания к самостоятельной работе.***

Самостоятельная работа обучающегося является основным средством овладения учебным материалом во время, свободное от обязательных учебных занятий. Самостоятельная работа обучающегося над усвоением учебного материала по учебной дисциплине может выполняться в библиотеке университета, учебных кабинетах, компьютерных классах, а также в домашних условиях. Содержание и количество самостоятельной работы обучающегося определяется учебной программой дисциплины, методическими материалами, практическими заданиями и указаниями преподавателя.

#### ***Самостоятельная работа в аудиторное время может включать:***

- 1) конспектирование (составление тезисов) лекций;
- 2) выполнение контрольных работ;
- 3) решение задач;
- 4) работу со справочной и методической литературой;
- 5) работу с нормативными правовыми актами;
- 6) выступления с докладами, сообщениями на семинарских занятиях;
- 7) защиту выполненных работ;
- 8) участие в оперативном (текущем) опросе по отдельным темам изучаемой дисциплины;

9) участие в беседах, деловых (ролевых) играх, дискуссиях, круглых столах, конференциях;

10) участие в тестировании и др.

***Самостоятельная работа во внеаудиторное время может состоять***

***из:***

1) повторения лекционного материала;

2) подготовки к практическим занятиям;

3) изучения учебной и научной литературы;

4) изучения нормативных правовых актов (в т.ч. в электронных базах данных);

5) решения задач, и иных практических заданий;

6) подготовки к контрольным работам, тестированию и т.д.;

7) подготовки к практическим занятиям устных докладов (сообщений);

8) выполнения курсовых работ, предусмотренных учебным планом;

9) выполнения выпускных квалификационных работ и др.;

10) выделения наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями на консультациях;

11) проведения самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов, докладов;

12) текущий контроль осуществляется в форме устных, тестовых опросов, докладов;

В случае пропусков занятий, наличия индивидуального графика обучения и для закрепления практических навыков студентам могут быть выданы типовые индивидуальные задания, которые должны быть сданы в установленный преподавателем срок.

### **13. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья**

Обучение по данной дисциплине инвалидов и лиц с ограниченными возможностями здоровья (далее ОВЗ) осуществляется преподавателем с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

Для студентов с нарушениями опорно-двигательной функции и с ОВЗ по слуху предусматривается сопровождение лекций и практических занятий мультимедийными средствами, раздаточным материалом.

Для студентов с ОВЗ по зрению предусматривается применение технических средств усиления остаточного зрения, а также предусмотрена возможность разработки аудиоматериалов.

По данной дисциплине обучение инвалидов и лиц с ограниченными возможностями здоровья может осуществляться как в аудитории, так и с использованием электронной информационно-образовательной среды, образовательного портала и электронной почты.

## ЛИСТ ДОПОЛНЕНИЙ И ИЗМЕНЕНИЙ

### рабочей программы дисциплины

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---