

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФАКУЛЬТЕТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Языки и методы программирования

Кафедра дискретной математики и информатики
факультета математики и компьютерных наук

Образовательная программа бакалавриата
01.03.02 – Прикладная математика и информатика

Направленность (профиль) программы
Математическое моделирование и вычислительная математика

Форма обучения

Очная

Статус дисциплины: входит в обязательную часть ОПОП

Махачкала, 2022

Рабочая программа дисциплины «Языки и методы программирования» составлена в 2022 году в соответствии с требованиями ФГОС ВО – бакалавриата по направлению подготовки 01.03.02 – Прикладная математика и информатика от 10.01.2018г. №9.

Разработчик(и): кафедра дискретной математики и информатики, ст. преподаватель Мирзабеков Я.М.

Рабочая программа дисциплины одобрена:
на заседании кафедры дискретной математики и информатики от 28.02.2022г,
протокол №6
Зав. кафедрой  Магомедов А.М.
(подпись)

на заседании Методического совета факультета математики и компьютерных наук от 24.03.2022г., протокол №4.

Председатель  Ризаев М.К.
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением « 31 » 03 2022г.

Начальник УМУ  Гасангаджиева А.Г.
(подпись)

Аннотация рабочей программы дисциплины

Дисциплина «Языки и методы программирования» входит в обязательную часть ОПОП бакалавриата по направлению 01.03.02 – Прикладная математика и информатика.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с современными знаниями в области языков и методов программирования.

Дисциплина нацелена на формирование следующих компетенций выпускника: общепрофессиональных – ОПК-4, ОПК-5, профессиональных – ПК-5.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции, практические занятия, лабораторные занятия и самостоятельная работа.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости в форме контрольной работы и промежуточный контроль в форме зачета и экзамена.

Объем дисциплины 8 зачетных единиц, в том числе в академических часах по видам учебных занятий

Объем дисциплины в очной форме

Семестр	Учебные занятия					Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)
	в том числе:					
	всего	Контактная работа обучающихся с преподавателем			СРС, в том числе экзамен	
		из них				
	Лекции	Лабораторные занятия	Практические занятия			
2	108	16	14	14	28+36	Экзамен
3	72	16	30		26	Зачет
4	108	14	28		30+36	Экзамен
итого	288	56	72	14	84+72	

1. Цели освоения дисциплины

Целью освоения дисциплины «Языки и методы программирования» является ознакомление студентов с историей развития языков программирования, с парадигмами программирования, структурами данных, знакомство с методами, применяемыми в программировании, известными алгоритмами, возможностью работы с динамической памятью.

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Языки и методы программирования» входит в обязательную часть ОПОП бакалавриата по направлению 01.03.02 – Прикладная математика и информатика.

Дисциплина «Языки и методы программирования» призвана содействовать знакомству студентов с языками и методами программирования и является курсом, для освоения которого необходимы теоретические знания и практические навыки, полученные по дисциплинам «Основы программирования», «Алгоритмы и анализ сложности», «Дискретная математика».

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения и процедура освоения).

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций (в соответствии с ОПОП)	Планируемые результаты обучения	Процедура освоения
ОПК-4 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ОПК-4.1 Знает основные положения и концепции прикладного и системного программирования, архитектуры компьютеров и сетей (в том числе и глобальных).	Знает: основные принципы документационного обеспечения профессиональной деятельности с учетом требований информационной безопасности; алгоритмы решения стандартных организационных задач; основные понятия, теоретические положения и методы программирования на языках высокого уровня. Умеет: применять методы программирования при решении разнообразных задач теоретического и практического содержания. Владеет: методами программирования на различных языках высокого уровня для решения теоретических и практических задач.	Конспектирование и проработка лекционного материала. Участие в лабораторных занятиях. Самостоятельная работа.

	<p>ОПК-4.2 Умеет использовать информационные технологии в профессиональной деятельности.</p>	<p>Знает: основные направления применения информационно-коммуникационных технологий в науке и образовании; принципы построения сетей; локальные и глобальные сети; сеть Интернет; безопасность компьютерных сетей. Умеет: выбирать эффективные информационные технологии для использования в научных исследованиях и учебном процессе. Владеет: методами математического и алгоритмического моделирования и информационно-коммуникационных технологий в науке и образовании.</p>	
	<p>ОПК-4.3 Имеет практические навыки разработки программного</p>	<p>Знает: теоретические положения и методы программирования на языках высокого уровня. Умеет: выбирать эффективные информационные технологии для использования в научных исследованиях и учебном процессе. Владеет: навыками построения алгоритмов и программ различных явлений и процессов, навыками использования информационных технологий для обработки данных.</p>	
<p>ОПК-5 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p>	<p>ОПК-5.1 Знает основные современные языки программирования, технологии создания и эксплуатации программных продуктов и программных комплексов.</p>	<p>Знает: основные принципы документационного обеспечения профессиональной деятельности с учетом требований информационной безопасности; алгоритмы решения стандартных организационных задач; основные понятия, теоретические положения и методы программирования на языках высокого уровня. Умеет: применять методы программирования при решении разнообразных задач</p>	<p>Конспектирование и проработка лекционного материала. Участие в лабораторных занятиях. Самостоятельная работа.</p>

		теоретического и практического содержания. Владеет: методами программирования на различных языках высокого уровня для решения теоретических и практических задач.	
	ОПК-5.2 Умеет разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения.	Знает: основные направления применения информационно-коммуникационных технологий в науке и образовании; принципы построения сетей; локальные и глобальные сети; сеть Интернет; безопасность компьютерных сетей. Умеет: выбирать эффективные информационные технологии для использования в научных исследованиях и учебном процессе. Владеет: методами математического и алгоритмического моделирования и информационно-коммуникационных технологий в науке и образовании.	
	ОПК-5.3 Имеет практические навыки разработки программного обеспечения.	Знает: теоретические положения и методы программирования на языках высокого уровня. Умеет: выбирать эффективные информационные технологии для использования в научных исследованиях и учебном процессе. Владеет: навыками построения алгоритмов и программ различных явлений и процессов, навыками использования информационных технологий для обработки данных.	
ПК-5 Способен к анализу требований к программному обеспечению	ПК-5.1 Знает методы анализа возможностей реализации требований к программному обеспечению	Знает: методы структурного анализа требований к программному обеспечению Умеет: применять методы разработки и исследования математических, информационных и имитационных моделей по	Конспектирование и проработка лекционного материала. Участие в лабораторных занятиях.

		<p>тематике выполняемых прикладных работ. Владеет: навыками разработки и исследования алгоритмов, протоколов, вычислительных моделей и баз данных для реализации функций и сервисов систем информационных технологий</p>	<p>Самостоятельная работа.</p>
	<p>ПК-5.2 Умеет использовать возможности существующей программно-технической архитектуры, методологию разработки программного обеспечения и технологии программирования</p>	<p>Знает: общие вопросы теории интеллектуальных систем, различные методы обработки информации, способы их программной реализации. Умеет: применять современные системные программные средства, технологии и инструментальные средства Владеет: основными методами, способами и средствами получения, хранения, переработки информации, иметь навыки работы с компьютером как средством управления информацией</p>	
	<p>ПК-5.3 Имеет практический опыт проведения оценки и обоснование рекомендуемых решений</p>	<p>Знает: архитектуру современных высокопроизводительных вычислительных систем Умеет: обеспечивать передачу информации между приложениями Владеет: навыками разработки проектной и программной документации; методикой разработки архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения.</p>	

4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 8 зачетных единиц, 288 академических часов.

4.2. Структура дисциплины.

4.2.1. Структура дисциплины в очной форме

№ п/п	Разделы и темы дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля успеваемости и промежуточной аттестации
			Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа	
Модуль 1.							
1	Python. Первые шаги.	2	4	2	2	4	Прием лабораторных и практических работ
2	Первые программы на Python.	2	2	2	2	4	Прием лабораторных и практических работ
3	Управляющие инструкции.	2	2	2	2	8	Прием лабораторных и практических работ
	<i>Итого по модулю 1:</i>		8	6	6	16	Контрольная работа
Модуль 2.							
1	Работа со списками, строками, множествами, словарями и кортежами в Python.	2	4	4	4	4	Прием лабораторных и практических работ
2	Библиотеки Python	2	4	4	4	8	Прием лабораторных и практических работ
	<i>Итого по модулю 2:</i>		8	8	8	12	Контрольная работа
Модуль 3.							
	Подготовка к экзамену	2				36	Экзамен
	<i>Итого за 2 сем</i>	2	16	14	14	28+36	
Модуль 1.							
1	Функции в языке Python.	3	4		6	6	Прием лабораторных и практических работ
2	Работа с файлами в Python.	3	4		8	8	Прием лабораторных и практических работ
	<i>Итого по модулю 1:</i>		8		14	14	Контрольная работа
Модуль 2.							

1	Основы ООП в Python.	3	4		8	6	Прием лабораторных и практических работ
2	Графический интерфейс программ в Python.	3	4		8	6	Прием лабораторных и практических работ
	<i>Итого по модулю 2:</i>		8		16	12	Зачет
	<i>Итого за 3 сем</i>	3	16		30	26	
Модуль 1.							
1	Работа с БД SQLite в Python	4	2		6	8	Прием лабораторных и практических работ
2	Создание простейших игр в Python	4	4		8	8	Прием лабораторных и практических работ
	<i>Итого по модулю 1:</i>		6		14	16	Контрольная работа
Модуль 2.							
1	Создание web-приложений в Python	4	6		8	8	Прием лабораторных и практических работ
2	Создание простейших чат-ботов в Python	4	2		6	6	Прием лабораторных и практических работ
	<i>Итого по модулю 2:</i>		8		14	14	Контрольная работа
Модуль 3.							
	Подготовка к экзамену	4				36	Экзамен
	<i>Итого на 4 сем</i>	4	14		28	30+36	
	ИТОГО:		56	14	72	84+72	

4.3. Содержание дисциплины, структурированное по темам (разделам).

4.3.1 Содержание лекционных занятий по дисциплине

2 семестр

Первый модуль

Тема 1. Python. Первые шаги.

Понятие «алгоритм» и «программа». Знакомство с Python. Вывод информации на экран. Ввод данных в программу.

Тема 2. Первые программы на Python.

Синтаксис языка Python. Структура программы. Переменные величины. Инструкция присваивания.

Тема 3. Управляющие инструкции.

Условный оператор if-elif-else. Функция range(). Операторы цикла for и while. Операторы break и continue. Обработка исключительных ситуаций.

Второй модуль

Тема 1. Работа со списками, строками, множествами, словарями и кортежами в Python.

Объекты типа set. Операции над множеством. Операции над двумя множествами. Сравнение множеств.

Строка как коллекция. Индексация в строках. Перебор элементов строки. Хранение текстов в памяти компьютера. Срезы строк.

Списки. Создание списков. Индексация в списках. Добавление элемента в список. Перебор элементов списка. Срезы списков. Удаление элементов.

Списки и массивы. Методы split и join. Списочные выражения. Использование списочных выражений в аргументах методов split и join. Считывание значений, введенных одной строкой. Вложенные списки. Двумерные вложенные списки (матрицы). Создание двумерного списка. Перебор элементов двумерного списка. Вывод списка на экран. Матрицы.

Кортежи. Присваивание кортежей. Преобразования между коллекциями.

Знакомство со словарями. Создание словаря. Обращение к элементу словаря. Добавление и удаление элементов. Проверка наличия элемента в словаре. Нестроковые ключи. Методы словарей. Допустимые типы ключей

Тема 2. Библиотеки Python

Библиотеки как наследие. Репозиторий PyPI. Встроенные модули. Модуль math. Модуль random. Модуль datetime. Модуль pprint.

Обработка изображений. Растровые изображения. PIL. Установка библиотек. Модельный пример. Создание изображений и рисование.

Работа с документами. Создание текстового документа. Работа с документом как с шаблоном. Создание презентаций. Создание таблиц Excel

3 семестр Первый модуль

Тема 1. Функции в языке Python.

Создание функции. Значения аргументов по умолчанию. Функция как аргумент. Рекурсия. Лямбда-функции. Локальные и глобальные переменные. Вложенные функции. Функция как результат функции.

Тема 2. Работа с файлами в Python.

Общие сведения о файлах. Перевод строки. Файловый путь. Относительные и абсолютные пути. Кодировки файлов. Типичные операции с файлами. Чтение файла. Запись в файл. Заккрытие файлов.

Второй модуль

Тема 1. Основы ООП в Python.

Основные понятия. Создание классов. Методы классов. Инициализация экземпляров класса. Полиморфизм. Проверка типа объекта. Иерархия классов. Механизм наследования.

Тема 2. Графический интерфейс программ в Python.

Графический интерфейс. Установка и настройка. Первые шаги в PyQt. Кто отправил сигнал. Открытие других форм. Установка QtDesigner и первый запуск. Подключение дизайна к программе. Размещение виджетов.

4 семестр Первый модуль

Тема 1. Работа с БД SQLite в Python

Введение в базы данных. Основы SQL. Работаем с SQLite базой данных из Python. Возможности PyQt по работе с базами данных. Модификация из Python.

Тема 2. Создание простейших игр в Python.

Установка pygame. Hello, Pygame. Рисование. Цвета. Система координат. Surface, Rect. Функции рисования. Время в PyGame. События. События по таймеру. Холст (Surface). Что такое клетчатое поле? Создание класса. Реакция поля на события мыши. Изображения. Спрайты. Наследование спрайтов. Пересечение по прямоугольнику и окружности. Пересечение по маске.

Второй модуль

Тема 1. Создание web-приложений в Python

Инструменты для разработки веб-приложений. Первая страница на Flask. Статический контент. Передача параметров в адресной строке. Обработка форм. Шаблоны. Условия в шаблонах. Циклы в шаблонах. Создание переменных в шаблоне. Наследование шаблонов. Знакомство с Flask-WTF. Базы данных и ORM. Установка. Постановка задачи для первого примера. Начальный каркас приложения. Таблица с пользователями. Таблица с новостями. Взаимодействие с базой данных. Небольшая шпаргалка по filter. Отображение публичных записей. Регистрация пользователей.

Тема 2. Создание простейших чат-ботов в Python.

Про чат-ботов. Выполнение запросов от имени пользователя. Загрузка файлов. Работа с событиями и чатами. Мессенджер Telegram. Как завести пользователя в Телеграме? API python-telegram-bot. Эхо-бот. Подключение через прокси-сервер. Обработка команд. Создание клавиатуры в диалоге пользователя. Установка и удаление таймера. Создание сценариев диалогов. Передача пользовательских данных в сценарии. Использование HTTP-API в телеграм-ботах.

**4.3.2 Содержание лабораторно-практических занятий по дисциплине
2 семестр
Первый модуль**

- Тема 1. Python. Первые шаги.**
- Тема 2. Первые программы на Python.**
- Тема 3. Управляющие инструкции.**

Второй модуль

- Тема 1. Работа со списками, строками, множествами, словарями и кортежами в Python.**
- Тема 2. Библиотеки Python**

**3 семестр
Первый модуль**

- Тема 1. Функции в языке Python.**
- Тема 2. Работа с файлами в Python.**

Второй модуль

- Тема 1. Основы ООП в Python.**
- Тема 2. Графический интерфейс программ в Python.**

**4 семестр
Первый модуль**

- Тема 1. Работа с БД SQLite в Python**
- Тема 2. Создание простейших игр в Python.**

Второй модуль

- Тема 1. Создание web-приложений в Python**
- Тема 2. Создание простейших чат-ботов в Python.**

5. Образовательные технологии

В процессе преподавания дисциплины «Языки и методы программирования» применяются следующие образовательные технологии: лекционно-зачетная система обучения, технология развития критического мышления, изложение теоретических положений в ходе лекционных занятий с применением современного интерактивного презентационного оборудования, проведение групповых дискуссий. Основная литература, презентации и конспекты лекций предоставляются студентам в электронном виде.

6. Учебно-методическое обеспечение самостоятельной работы студентов.

Самостоятельная работа студентов складывается из:

- проработки лекционного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных на лекциях упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к отчетам по лабораторным работам;
- подготовки к сдаче промежуточных форм контроля.

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
1.	Проработка лекционного материала	Контрольный фронтальный опрос	См. разделы 7.1, 8, 9 данного документа
2.	Изучение рекомендованной литературы и материалов соответствующих форумов интернет	Контрольный фронтальный опрос, прием и представление рефератов.	См. разделы 7.1, 8, 9 данного документа
3.	Подготовка к отчетам по практическим работам.	Проверка выполнения работ, опрос по теме работы.	См. разделы 7.1, 8, 9 данного документа
4.	Подготовка к сдаче промежуточных форм контроля	Контрольные работы по каждому модулю.	См. разделы 7.1, 8, 9 данного документа

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1. Типовые контрольные задания

Вопросы к зачету и экзамену

1. Переменные и типы данных языка Python. Преобразование типов
2. Ввод-вывод данных, оператор присваивания
3. Условный оператор
4. Функция range()

5. Операторы цикла
6. Функции
7. Лямбда-функции
8. Функции с переменным числом аргументов
9. Списки
10. Вложенные списки
11. Методы split и join. Списочные выражения
12. Срезы (строки, списки)
13. Кортежи
14. Множества
15. Словари
16. Файлы
17. Библиотеки Python. Модули math и random
18. Классы в Python. Наследование классов.
19. Создание приложений с графическим интерфейсом. Модуль PyQt5.
20. БД SQLite в Python
21. Модуль PyGame. Создание игр.
22. Модуль Flask. Создание web-приложений.
23. Чат-боты.

Варианты тестовых заданий

1. Для чего предназначено окно Инспектора объектов?
 - a) для отображения на нем и изменения свойств выбранных компонентов и событий, на которые будет реагировать эта форма или ее компоненты;
 - b) для изменения свойств выбранных компонентов и событий, на которые будет реагировать эта форма или ее компоненты, создания обработчиков событий;
 - c) для декоративных целей;
 - d) проектируемого Windows-приложения.
2. Какое расширение имеет файл проекта в Delphi?
 - a) dpr;
 - b) pas;
 - c) dfm;
 - d) exe.
3. Выполнение какой операции обеспечивает функция StrToFloat?
 - a) обеспечивает преобразование строкового изображения числа в его значение действительного типа;
 - b) обеспечивает преобразование действительного числа в его символьное изображение;
 - c) обеспечивает преобразование действительного числа в его двоичное представление.
4. Первоначальный код головной программы и отдельных модулей создается:
 - a) автоматически системой Delphi;

b) разработчиком программы;
c) автоматически системой Delphi, далее он недоступен для редактирования разработчиком.

5. Совпадает ли имя проекта с именем файла, в котором он сохранен?

- a) имя проекта совпадает с именем файла, в котором он сохранен;
- b) нет;
- c) не обязательно.

6. Может ли вводиться комментарий в тексте программы с помощью сочетаний символов «(*)» и «(*)»?

- a) да;
- b) нет;
- c) с разрешения преподавателя.

7. Какую информацию содержит файл проекта в среде Delphi?

- a) содержит исходный код главной подпрограммы;
- b) информацию об объектах формы и их свойствах;
- c) процедуры обработки событий, связанных с нажатием кнопок.

8. Какую информацию содержит файл формы в среде Delphi?

- a) в нем перечислены объекты формы и значения свойств этих объектов;
- b) информацию о формах компонент и их свойствах;
- c) перечислены все параметры компонентов и их свойства.

9. Какую информацию содержит файл модуля в среде Delphi?

- a) исходный код модуля, ассоциированного с одноименной формой;
- b) процедуры обработки событий, связанных с нажатием кнопок, и описания глобальных переменных;
- c) информацию о формах объектов, компонент и их свойствах.

10. Для создания несложных пользовательских интерфейсов чаще всего используются следующие простые компоненты Delphi:

метки, поля ввода, области просмотра, кнопки. На какой странице Палитры компонентов они находятся?

- a) во вкладке Samples Палитры компонентов;
- b) во вкладке Standard Палитры компонентов;
- c) во вкладке Additional Палитры компонентов.

11. В результате выполнения оператора цикла
for i: = 1 to 50 do If x[i]>0 then x[i]:= x[i]* x[i]; :

- a) возведутся в квадрат положительные элементы массива x;
- b) перемножатся все элементы массива x;
- c) перемножатся отрицательные элементы массива x.

12. Массив – это:

- a) совокупность однотипных данных, имеющих общее имя;
- b) совокупность строк и столбцов;
- c) совокупность разнотипных данных, имеющая общее имя.

13. Выберите правильное обозначение элемента массива в Pascal:

- a) a[i];
- b) a[5];

- c) a(i);
- d) a(3);
- e) a{i}.

14. Выберите правильные варианты оператора объявления массива:

- a) var a: array[1..25] of Integer;
- b) var a: array(1..25) of Integer;
- c) var a: array[1..25] of real;
- d) var a: array[1...25] of real.

15. Найти ошибку в фрагменте программы:

```
var i: Integer;  
var s: Real;  
var a: array[1..25] of Real;  
begin  
s:=0;  
for i:= 1 to 40 do s :=s+a[i];  
end;
```

- a) значения параметра цикла i (от 26 до 40) выходят за пределы размерности объявленного массива [1..25];
- b) ошибка в операторе объявления переменной s;
- c) нарушение структуры программы.

16. Какие из приведенных констант относятся к строковому типу?

- a) 'CONST';
- b) '5467';
- c) 5467;
- d) '1'.

17. Каким ключевым словом задается в разделе описания переменных строковый тип?

- a) char;
- b) integer;
- c) byte;
- d) string.

18. Какое значение примет переменная k в результате выполнения фрагмента программы?

```
m := 'биология';  
k := 'зоо' + сору (m, 4, 5);
```

- a) зоолог;
- b) зоогия;
- c) зоология;
- d) биология.

19. Какое значение примет переменная a в результате выполнения фрагмента программы?

```
a := 'комар';  
b := 'ово';  
insert (b, a, 6);  
a) комар;
```

- b) комарово;
- c) корова;
- d) ово.

20. Какой результат может дать строковая функция?

- a) переменная;
- b) число;
- c) символ;
- d) строка.

7.2. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Общий результат выводится как интегральная оценка, складывающаяся из текущей работы - 50% и текущего контроля - 50%.

Текущая работа по дисциплине включает:

- посещение занятий - 10 баллов,
- участие на практических занятиях - 20 баллов,
- выполнение лабораторных заданий - 30 баллов,
- выполнение домашних (аудиторных) контрольных работ - 40 баллов.

Текущий контроль по дисциплине включает:

- устный опрос - 40 баллов,
- письменная контрольная работа - 60 баллов.

Промежуточный контроль по дисциплине включает экзамен, результаты которого оцениваются по 100-балльной системе ориентировочно по следующим критериям:

- 1) оценка «отлично», если у студента от 86 до 100 баллов с учетом степени усвоения, высокий уровень знаний по программе дисциплины, отвечает четко и логически обоснованно;
- 2) оценка «хорошо», если у студента от 66 до 85 баллов с учетом степени усвоения, достаточно высокий уровень знаний по программе дисциплины, отвечает в основном четко и логически обоснованно, но допускает отдельные неточности.
- 3) оценка «удовлетворительно», если у студента от 51 до 65 баллов с учетом степени усвоения, достаточный уровень знаний по программе дисциплины, отвечает в основном правильно и в логической последовательности, но допускает отдельные неточности;
- 4) оценка «неудовлетворительно», если у студента от 0 до 50 баллов с учетом степени усвоения, недостаточный уровень знаний по программе дисциплины, имеются существенные пробелы в усвоении важных знаний из программы курса.

8. Учебно-методическое обеспечение дисциплины.

а) адрес сайта курса

<http://edu.dgu.ru/course/view.php?id=934> – Курс: Языки и методы программирования

б) основная литература:

1. Павловская Т.А. С/С++. Программирование на языке высокого уровня: для магистров и бакалавров / Павловская, Татьяна Александровна. - СПб. [и др.]: Питер, 2012. - 460 с. - (Учебник для вузов). - Допущено МО РФ. - ISBN 978-5-94723-568-5: 357-00.
Местонахождение: Научная библиотека ДГУ
2. Сузи Р.А. Язык программирования Python: курс / Р.А. Сузи. - 2-е изд., испр. - Москва: Интернет-Университет Информационных Технологий, 2007. - 327 с. - (Основы информационных технологий). - ISBN 978-5-9556-0109-0; То же [Электронный ресурс]. -
URL: <http://biblioclub.ru/index.php?page=book&id=233288>
3. Северенс, Ч. Введение в программирование на Python / Ч. Северенс. - 2-е изд., испр. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 231 с.: схем., ил.; То же [Электронный ресурс]. -
URL: <http://biblioclub.ru/index.php?page=book&id=429184>

б) дополнительная литература:

1. Delphi: Программирование на языке высокого уровня: учеб. для студентов вузов, обуч. по направлению подгот. диплом. специалистов "Информ. и вычисл. техн." / Фаронов, Валерий Васильевич. - СПб. [и др.]: Питер, 2008. - 639 с. - (Учебник для вузов). - Допущено МО РФ. - ISBN 978-5-8046-0008-3:260-00.
Местонахождение: Научная библиотека ДГУ
2. Буйначев, С.К. Основы программирования на языке Python: учебное пособие / С.К. Буйначев, Н.Ю. Боклаг; Министерство образования и науки Российской Федерации, Уральский федеральный университет им. первого Президента России Б. Н. Ельцина. - Екатеринбург: Издательство Уральского университета, 2014. - 92 с.: табл., ил. - Библиогр. в кн. - ISBN 978-5-7996-1198-9; То же [Электронный ресурс]. -
URL: <http://biblioclub.ru/index.php?page=book&id=275962>
3. Sweigart, A. Разработка компьютерных игр на языке Python / A. Sweigart. - 2-е изд., испр. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 505 с.: ил.; То же [Электронный ресурс]. -
URL: <http://biblioclub.ru/index.php?page=book&id=429009>

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.

1) eLIBRARY.RU [Электронный ресурс]: электронная библиотека / Науч. электрон. б-ка. — Москва, 1999 —. Режим доступа:

<http://elibrary.ru/defaultx.asp> – Яз. рус., англ.

2) Moodle [Электронный ресурс]: система виртуального обучением: [база данных] / Даг. гос. ун-т. – Махачкала, г. – Доступ из сети ДГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. –

URL: <http://edu.dgu.ru/>

3) Электронный каталог НБ ДГУ [Электронный ресурс]: база данных содержит сведения о всех видах лит, поступающих в фонд НБ

ДГУ/Дагестанский гос. ун-т. – Махачкала, 2010 – Режим доступа:

<http://elib.dgu.ru>, свободный.

4) <https://pythontutor.ru/> - Бесплатный курс по программированию с нуля.

10. Методические указания для обучающихся по освоению дисциплины.

При решении лабораторных заданий программистский подход непременно должен присутствовать (без него решение не будет полноценным), однако, он не должен заслонять сугубо математические (доказательство и др.) и алгоритмические (построение, оптимизация, верификация и др.) аспекты.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение: Microsoft Windows, Python 3.10, Pycharm 2022 Community. Также студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.

Имеется необходимая литература в библиотеке, медиапроектор и компьютер для проведения лекций-презентаций.

Лабораторные занятия проводятся в компьютерных классах с необходимым программным обеспечением.