



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОДЕЗИИ И КАРТОГРАФИИ
(МИИГАиК)

Факультет архитектуры и градостроительства
Кафедра информатики и геоинформационных технологий



**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

Сертификат: 5387FD2AF5B17FED816FDE9A1D4973B1
Владелец: Степанченко Алексей Леонидович
Действителен: с 21.07.2025 по 14.10.2026

«УТВЕРЖДАЮ»

Проректор
А. Л. Степанченко
24 марта 2025 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)
«Б1.В.ДВ.07.04 ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON»**

Уровень высшего образования: бакалавриат

Направление подготовки: 07.03.01 Архитектура

Направленность(профиль) подготовки: Архитектура

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Год набора (приема на обучение): 2025

Срок получения образования: 5 лет

Объем:
в зачетных единицах: 2 з.е.
в академических часах: 72 ак.ч.

Москва, 2025

Разработчики:

Старший преподаватель кафедры Информатики и геоинформационных технологий Стрельцов А. С.

Фонд оценочных средств по дисциплине (модулю) составлен в соответствии с требованиями ФГОС ВО по направлению подготовки 07.03.01 Архитектура, утвержденного приказом Минобрнауки от 08.06.2017 № 509

Согласование и утверждение

№	Подразделение	Ответственное лицо	ФИО	Виза	Дата, протокол (при наличии)
1	факультет Архитектуры и градостроительств а	Председатель ученого совета факультета	Хачатрян К. О.	Рассмотрено	19.12.2024, № 3-2024/2025

1. Планируемые результаты обучения, соотнесенные с планируемыми результатами освоения образовательной программы

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-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения

Знать:

УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач

Уметь:

УК-2.2/Ум15 Писать и запускать простые программы на языке Python

Владеть:

УК-2.2/Нв14 Навыками разработки и отладки простых программ на Python

УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач

Знать:

УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами

Уметь:

УК-2.5/Ум15 Использовать условия, циклы, функции и структуры данных для решения учебных задач

Владеть:

УК-2.5/Нв10 Навыком считывать, обрабатывать и сохранять простые текстовые и табличные данные

2. Показатели и критерии оценивания компетенций

Планируемые результаты освоения программы	Уровень овладения	Критерии оценивания (перечень сформированных ИДК/ЗУВ по уровням освоения)
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	Повышенный	ИДК УК-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения Знать: УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач Уметь: УК-2.2/Ум15 Писать и запускать простые программы на языке Python Владеть: УК-2.2/Нв14 Навыками разработки и отладки простых программ на Python

	ИДК УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач Знать: УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами Уметь: УК-2.5/Ум15 Использовать условия, циклы, функции и структуры данных для решения учебных задач Владеть: УК-2.5/Нв10 Навыком считывать, обрабатывать и сохранять простые текстовые и табличные данные
Базовый	ИДК УК-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения Знать: УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач Уметь: УК-2.2/Ум15 Писать и запускать простые программы на языке Python ИДК УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач Знать: УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами Уметь: УК-2.5/Ум15 Использовать условия, циклы, функции и структуры данных для решения учебных задач
Пороговый	ИДК УК-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения Знать: УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач

		ИДК УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач Знать: УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами
--	--	---

3. Шкала оценивания

Оценка	Критерии оценивания (перечень сформированных ИДК/ЗУВ)
Отлично	УК-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач УК-2.2/Ум15 Писать и запускать простые программы на языке Python УК-2.2/Нв14 Навыками разработки и отладки простых программ на Python УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами УК-2.5/Ум15 Использовать условия, циклы, функции и структуры данных для решения учебных задач УК-2.5/Нв10 Навыком считывать, обрабатывать и сохранять простые текстовые и табличные данные
Хорошо	УК-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач УК-2.2/Ум15 Писать и запускать простые программы на языке Python УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами УК-2.5/Ум15 Использовать условия, циклы, функции и структуры данных для решения учебных задач
Удовлетворительно	УК-2.2 Определяет связи между поставленными задачами и ожидаемые результаты их решения УК-2.2/Зн15 Основные возможности Python и областей его применения для решения прикладных задач

	УК-2.5 Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач и выбирает оптимальный способ решения поставленных задач УК-2.5/Зн11 Базовые типы данных, переменных, операций ввода и вывода, условных операторов и циклов, основные структуры данных, функций и простые способы работы с файлами
Неудовлетворительно	Результаты обучения не сформированы на достаточном уровне

4. Контрольные мероприятия по дисциплине

Вид контроля	Форма контроля/Оценочное средство
Текущий контроль	Тестирование
Промежуточная аттестация	Зачет с оценкой

№ п/п	Наименование раздела	Контролируемые ИДК	Вид контроля/ используемые оценочные материалы	
			Текущий	Промежут. аттестация
1	Основы программирования и среда разработки	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
2	Линейные алгоритмы и базовые операции	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
3	Управляющие конструкции	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
4	Обработка строк и текстовых данных	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
5	Структуры данных	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
6	Функциональное программирование	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
7	Модульная организация программ	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
8	Объектно-ориентированное программирование	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
9	Итераторы и функциональные возможности	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой
10	Работа с файловой системой	УК-2.2 УК-2.5	Тестирование	Зачет с оценкой

5. Порядок проведения промежуточной аттестации

Порядок проведения промежуточной аттестации регламентируется локальными нормативными актами Университета:

- Положением о формах, периодичности, порядке текущего контроля успеваемости и промежуточной аттестации обучающихся по основным образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры в федеральном государственном бюджетном образовательном учреждении высшего образования «Московский государственный университет геодезии и картографии», п.6 «Организация и проведение промежуточной аттестации».
- Положением о порядке осуществления образовательного процесса, проведения текущего контроля успеваемости и промежуточной аттестации обучающихся с применением электронного обучения и дистанционных образовательных технологий в федеральном государственном бюджетном образовательном учреждении высшего образования «Московский государственный университет геодезии и картографии», п.8 «Проведение промежуточной аттестации с применением ЭО и ДОТ».

Промежуточная аттестация: Зачет с оценкой, 5 семестр.

Зачет с оценкой проводится в устно-практической форме с использованием компьютерной техники. В начале студент получает билет, содержащий один теоретический вопрос и две практические задачи различного уровня сложности - базового и повышенного. После получения билета студент приступает к решению практической задачи базового уровня на компьютере в среде разработки Python, параллельно с выполнением задания готовя ответ на теоретический вопрос билета.

По завершении решения задачи базового уровня преподаватель проводит промежуточную проверку, оценивая корректность программной реализации, качество кода и соответствие решения поставленным требованиям. Затем студент устно отвечает на теоретический вопрос билета, после чего преподаватель при необходимости задает дополнительные вопросы по теме билета и смежным разделам дисциплины. После успешного прохождения промежуточной аттестации студенту выдается дополнительный бланк с практическим заданием повышенной сложности для выполнения на компьютере.

Итоговая оценка за зачет выставляется на основании комплексной проверки теоретических знаний и практических навыков программирования в соответствии с критериями оценивания, установленными в фонде оценочных средств дисциплины.

6. Оценочные материалы текущего контроля

Раздел 1. Основы программирования и среда разработки

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 1.1. Введение в язык программирования Python

Форма контроля/оценочное средство: Тестирование

Тема 1.2. Интегрированная среда разработки

Форма контроля/оценочное средство: Тестирование

Тема 1.3. Базовые элементы языка

Форма контроля/оценочное средство: Тестирование

Раздел 2. Линейные алгоритмы и базовые операции

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 2.1. Функции ввода-вывода

Форма контроля/оценочное средство: Тестирование

Тема 2.2. Арифметические операции и математические функции

Форма контроля/оценочное средство: Тестирование

Тема 2.3. Генерация случайных чисел

Форма контроля/оценочное средство: Тестирование

Раздел 3. Управляющие конструкции

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 3.1. Условные операторы

Форма контроля/оценочное средство: Тестирование

Тема 3.2. Циклические конструкции

Форма контроля/оценочное средство: Тестирование

Тема 3.3. Вложенные циклы и сложные алгоритмы

Форма контроля/оценочное средство: Тестирование

Раздел 4. Обработка строк и текстовых данных

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 4.1. Строковый тип данных

Форма контроля/оценочное средство: Тестирование

Тема 4.2. Операции со строками

Форма контроля/оценочное средство: Тестирование

Тема 4.3. Форматирование и регулярные выражения

Форма контроля/оценочное средство: Тестирование

Раздел 5. Структуры данных

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 5.1. Списки и операции над ними

Форма контроля/оценочное средство: Тестирование

Тема 5.2. Кортежи и множества

Форма контроля/оценочное средство: Тестирование

Тема 5.3. Словари и их применение

Форма контроля/оценочное средство: Тестирование

Раздел 6. Функциональное программирование

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 6.1. Определение и вызов функций

Форма контроля/оценочное средство: Тестирование

Тема 6.2. Расширенные возможности функций

Форма контроля/оценочное средство: Тестирование

Тема 6.3. Функции высшего порядка и декораторы

Форма контроля/оценочное средство: Тестирование

Раздел 7. Модульная организация программ

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 7.1. Работа с модулями

Форма контроля/оценочное средство: Тестирование

Тема 7.2. Пакеты и их структура

Форма контроля/оценочное средство: Тестирование

Тема 7.3. Стандартная библиотека

Форма контроля/оценочное средство: Тестирование

Раздел 8. Объектно-ориентированное программирование

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 8.1. Основы ООП

Форма контроля/оценочное средство: Тестирование

Тема 8.2. Наследование и полиморфизм

Форма контроля/оценочное средство: Тестирование

Тема 8.3. Специальные методы и инкапсуляция

Форма контроля/оценочное средство: Тестирование

Раздел 9. Итераторы и функциональные возможности

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 9.1. Протокол итерации

Форма контроля/оценочное средство: Тестирование

Тема 9.2. Контейнерные типы

Форма контроля/оценочное средство: Тестирование

Тема 9.3. Перечисления и именованные константы

Форма контроля/оценочное средство: Тестирование

Раздел 10. Работа с файловой системой

Контролируемые ИДК: УК-2.2 УК-2.5

Тема 10.1. Файловый ввод-вывод

Форма контроля/оценочное средство: Тестирование

Тема 10.2. Операции с файлами и каталогами

Форма контроля/оценочное средство: Тестирование

Тема 10.3. Сериализация данных

7. Оценочные материалы промежуточной аттестации

5 семестр, Зачет с оценкой

Контролируемые ИДК: УК-2.2 УК-2.5

Вопросы/Задания:

1. Какой тип данных в Python является неизменяемым (immutable)?
 - A. list
 - B. dict
 - C. set
 - D. tuple
2. Что выведет следующий код: `print(type(5/2))`?
 - A. `<class 'int'>`
 - B. `<class 'float'>`
 - C. `<class 'double'>`
 - D. `<class 'number'>`
3. Какой оператор используется для целочисленного деления в Python?
 - A. `/`
 - B. `//`
 - C. `%`
 - D. `div`
4. Какой метод используется для добавления элемента в конец списка?
 - A. `add()`
 - B. `insert()`
 - C. `append()`
 - D. `push()`
5. Что выведет код: `print("Hello"[1:4])`?
 - A. Hel
 - B. ell
 - C. ello
 - D. Hell
6. Какое ключевое слово используется для определения функции в Python?
 - A. `function`
 - B. `def`
 - C. `func`
 - D. `define`
7. Что делает оператор `continue` в цикле?
 - A. Завершает выполнение цикла
 - B. Переходит к следующей итерации цикла
 - C. Приостанавливает выполнение цикла
 - D. Возвращает значение из цикла
8. Какой модуль используется для работы с регулярными выражениями?
 - A. `regex`
 - B. `regexp`
 - C. `re`
 - D. `pattern`
9. Что означает `*args` в определении функции?
 - A. Обязательный аргумент
 - B. Произвольное количество позиционных аргументов
 - C. Именованный аргумент

D. Аргумент по умолчанию

10. Какой метод используется для удаления пробелов в начале и конце строки?

- A. remove()
- B. delete()
- C. strip()
- D. clean()

11. Что выведет код: `print(len([1, 2, [3, 4]]))`?

- A. 4
- B. 3
- C. 2
- D. 5

12. Какая функция используется для генерации последовательности чисел?

- A. seq()
- B. sequence()
- C. range()
- D. numbers()

13. Что такое lambda-функция в Python?

- A. Функция для работы со списками
- B. Анонимная функция, определяемая в одну строку
- C. Функция для математических вычислений
- D. Функция высшего порядка

14. Какой оператор используется для проверки вхождения элемента в последовательность?

- A. has
- B. contains
- C. in
- D. exists

15. Какое исключение возникает при делении на ноль?

- A. ValueError
- B. TypeError
- C. ZeroDivisionError
- D. MathError

16. Какой метод используется для преобразования строки в список?

- A. list()
- B. split()
- C. tolist()
- D. array()

17. Что такое множество (set) в Python?

- A. Упорядоченная коллекция элементов
- B. Неупорядоченная коллекция уникальных элементов
- C. Список с фиксированным размером
- D. Словарь без значений

18. Какой оператор используется для остатка от деления?

- A. //
- B. /
- C. %
- D. mod

19. Какой метод используется для объединения списков?

- A. merge()
- B. concat()
- C. extend()
- D. combine()

20. Что такое декоратор в Python?

- A. Функция, которая модифицирует поведение другой функции
- B. Оформление кода
- C. Комментарий к функции
- D. Тип переменной

21. Какой модуль используется для работы с JSON?

- A. jsonlib
- B. json
- C. simplejson
- D. data

22. Что делает оператор with при работе с файлами?

- A. Открывает файл для записи
- B. Автоматически закрывает файл после выполнения блока кода
- C. Создает новый файл
- D. Читает содержимое файла

23. Какой метод используется для получения всех ключей словаря?

- A. get_keys()
- B. keys()
- C. all_keys()
- D. list_keys()

24. Что означает self в методах класса?

- A. Статический метод
- B. Ссылка на текущий экземпляр класса
- C. Приватная переменная
- D. Конструктор класса

25. Какой модуль используется для генерации случайных чисел?

- A. rand
- B. random
- C. randint
- D. rng

26. Что выведет код: `print([1, 2, 3] * 2)`?

- A. [2, 4, 6]
- B. [1, 2, 3, 1, 2, 3]
- C. 6
- D. [1, 2, 3, 2]