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

Факультет оптического приборостроения
Кафедра информационно-измерительных систем

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

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

«УТВЕРЖДАЮ»
Проректор
А. Л. Степанченко
15 мая 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ) «Б1.О.16 ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

Уровень высшего образования: специалитет

Специальность: 12.05.01 Электронные и оптико-электронные приборы и системы специального назначения

Направленность(профиль) подготовки: Оптико-электронные информационно-измерительные приборы и системы

Квалификация (степень) выпускника: инженер

Форма обучения: очная

Год набора (приема на обучение): 2026

Срок получения образования: 5 лет 6 месяца(-ев)

Объем: в зачетных единицах: 4 з.е.
в академических часах: 144 ак.ч.

Москва, 2026

Разработчики:

Преподаватель кафедры Информационно-измерительных систем
Рамазанов Э. Р.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по специальности 12.05.01 Электронные и оптико-электронные приборы и системы специального назначения, утвержденного приказом Минобрнауки от 09.08.2018 № 93

Согласование и утверждение

№	Подразделение	Ответственное лицо	ФИО	Виза	Дата, протокол (при наличии)
1	факультет Оптического приборостроения	Председатель ученого совета факультета	Торшина И. П.	Рассмотрено	10.02.2026, № 5/25-26

Содержание

1. Цель и задачи освоения дисциплины (модуля)

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

3. Место дисциплины в структуре ОП

4. Объем дисциплины и виды учебной работы

5. Содержание дисциплины
   5.1. Разделы, темы дисциплины и виды занятий
   5.2. Содержание разделов, тем дисциплины и формы текущего контроля

6. Оценочные материалы текущего контроля

7. Оценочные материалы промежуточной аттестации

8. Порядок проведения промежуточной аттестации

9. Материально-техническое и учебно-методическое обеспечение дисциплины
   9.1. Перечень основной и дополнительной учебной литературы
   9.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся
   9.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине
   9.4. Оборудование и технические средства обучения

10. Методические указания по освоению дисциплины (модуля)

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - приобретение обучающимися знаний в области структурного программирования, расширение практических навыков программирования на языке Python с использованием стандартных средств разработки.

Задачи изучения дисциплины:

- знакомство студентов с основными видами языков программирования высокого уровня;
- формирование навыков работы с различными средствами программирования и отладки для создания программного обеспечения на языках высокого уровня;
- обучение основным принципам алгоритмического подхода, от этапа формализации до реализации в виде программного кода.

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Компетенции, индикаторы и результаты обучения

ОПК-3 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности

ОПК-3.1 Использует современные информационные технологии и программное обеспечение при решении задач профессиональной деятельности.

Знать:

ОПК-3.1/Зн1 Принципы построения и основные компоненты современного компьютера

ОПК-3.1/Зн2 Представление числовой, текстовой и графической информации в компьютере

Уметь:

ОПК-3.1/Ум5 Применять программное обеспечение при решении задач профессиональной деятельности

Владеть:

ОПК-3.1/Нв5 Навыками разработки алгоритмов программ с использованием структурного (императивного) объектно-ориентированного подходов в программировании

3. Место дисциплины в структуре ОП

Дисциплина (модуль) Б1.О.16 «Языки программирования» относится к обязательной части образовательной программы и изучается в семестре(ах): 3.

В процессе изучения дисциплины студент готовится к решению типов задач профессиональной деятельности, предусмотренных ФГОС ВО и образовательной программой.

4. Объем дисциплины (модуля) и виды учебной работы

Период	удоемкость сы)	удоемкость ЗЕТ)	ые занятия сы)	кие занятия сы)	льная работа сы)	ая аттестация сы)

обучения	Общая тру (ча	Общая тру (ЗЕ	Лекционны (ча	Практическ (ча	Самостоятел (ча	Промежуточн (ча
3 семестр	144	4	34	34	76	Зачет с оценкой
Всего	144	4	34	34	76	

5. Содержание дисциплины (модуля)

5.1. Разделы, темы дисциплины и виды занятий

(часы промежуточной аттестации не указываются)

Наименование раздела, темы	Всего	Лекционные занятия	Практические занятия	Самостоятельная работа	Планируемые результаты обучения, соотнесенные с результатами освоения программы
Раздел 1. Введение в программирование	**10**	**2**	**4**	**4**	ОПК-3.1
Тема 1.1. Введение в программирование	10	2	4	4	
Раздел 2. Синтаксис Python	**24**	**6**	**6**	**12**	ОПК-3.1
Тема 2.1. Синтаксис Python	24	6	6	12	
Раздел 3. Классы, интерфейсы, пакеты	**30**	**8**	**6**	**16**	ОПК-3.1
Тема 3.1. Классы, интерфейсы, пакеты	30	8	6	16	
Раздел 4. Обработка исключений	**28**	**6**	**6**	**16**	ОПК-3.1
Тема 4.1. Обработка исключений	28	6	6	16	
Раздел 5. Ввод-вывод	**28**	**6**	**6**	**16**	ОПК-3.1
Тема 5.1. Ввод-вывод	28	6	6	16	
Раздел 6. Параметризация типов в Python	**24**	**6**	**6**	**12**	ОПК-3.1
Тема 6.1. Параметризация типов в Python	24	6	6	12	
Итого	**144**	**34**	**34**	**76**	

5.2. Содержание разделов, тем дисциплин и формы текущего контроля

Раздел 1. Введение в программирование

Тема 1.1. Введение в программирование

Методологические основы процесса решения задач на ПК. Виды алгоритмов. Основные алгоритмические структуры. Алгоритм разветвляющейся структуры. Алгоритм циклической структуры. Алгоритмы цикла с предусловием, цикла с постусловием. Алгоритм цикла со счетчиком.

Текущий контроль

Вид (форма) контроля
Тестирование

Раздел 2. Синтаксис Python

Тема 2.1. Синтаксис Python

Основные понятия программирования на языке Python. Принципы работы с интерпретатором языка. Структура программы. Типы данных, операции, операторы.

Текущий контроль

Вид (форма) контроля
Тестирование

Раздел 3. Классы, интерфейсы, пакеты

Тема 3.1. Классы, интерфейсы, пакеты

Классы. Описание класса. Описание объектов. Конструкторы объектов. Статические элементы класса. Дружественные функции. Деструкторы.

Текущий контроль

Вид (форма) контроля
Тестирование

Раздел 4. Обработка исключений

Тема 4.1. Обработка исключений

Наследование. Шаблоны классов и обработка исключительных ситуаций.

Текущий контроль

Вид (форма) контроля
Тестирование

Раздел 5. Ввод-вывод

Тема 5.1. Ввод-вывод

Основные операторы циклического выполнения в языке Python. Вложенные циклические структуры. операторы прерывания цикла. Функция в языке Python. Определение пользовательских функций. Файловый ввод -вывод. Чтение строк с помощью итераторов файлов.

Текущий контроль

Вид (форма) контроля
Тестирование

Раздел 6. Параметризация типов в Python

Тема 6.1. Параметризация типов в Python

Формы наследования. Формы вложения.

Текущий контроль

Вид (форма) контроля
Тестирование

6. 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, 3 семестр.

Зачет оценивается по пятибалльной шкале:

- зачтено с оценкой «отлично» выставляется, если обучающийся имеет глубокие, исчерпывающие знания в объеме освоенной программы; знание основной (обязательной) литературы; дает правильные и уверенные ответы, свидетельствующие о наличии твердых знаний и навыков в использовании технических средств; полное, четкое, грамотное и логически стройное изложение материала; демонстрирует свободное применение теоретических знаний при анализе практических вопросов;
- зачтено с оценкой «хорошо» выставляется обучающемуся в случае соблюдения тех же требований, но наличии в ответе обучающегося по некоторым перечисленным показателям недостатков принципиального характера, что вызывает замечания или поправки преподавателя;
- зачтено с оценкой «удовлетворительно» выставляется обучающемуся в случае соблюдения тех же требований, но в случае наличия в ответе ошибок, необходимости существенной помощи в виде поправок и наводящих вопросов преподавателя;
- не зачтено («не удовлетворительно») – обучающийся не смог ответить на 2/3 вопросов; не справился с практическим заданием или выполнено менее 50 % практических заданий.

7. Оценочные материалы текущего контроля

Раздел 1. Введение в программирование

Контролируемые ИДК: ОПК-3.1

Тема 1.1. Введение в программирование

Форма контроля/оценочное средство: Тестирование

Вопросы/Задания:

1. Какой результат выведет следующий код?

```
x = 5
y = "5"
print(x == y)
```

1) True
2) False
3) Error
4) None

2. Что такое изменяемый (mutable) тип данных в Python?

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

3. Какой индекс у последнего элемента списка [1, 2, 3, 4, 5] ?

1) 5
2) 4
3) -1
4) Оба варианта b и c верны

4. Что выведет print("Hello"[1:4]) ?

1) "Hel"
2) "ell"
3) "ello"
4) "Hell"

5. Какой оператор используется для проверки принадлежности элемента списку?

1) ==
2) is
3) in
4) has

Раздел 2. Синтаксис Python

Контролируемые ИДК: ОПК-3.1

Тема 2.1. Синтаксис Python

Форма контроля/оценочное средство: Тестирование

Вопросы/Задания:

1. Что выведет следующий код?

```
for i in range(3):
    print(i)
```

1) 1 2 3
2) 0 1 2
3) 1 2
4) 0 1 2 3

2. Как правильно определить функцию в Python?

1) function myFunc():
2) def myFunc():
3) func myFunc():
4) define myFunc():

3. Что выведет print(len("Python")) ?

1) 5
2) 6
3) 7
4) Error

4. Какой тип данных возвращает функция input() ?

1) int
2) float
3) str
4) bool

5. Что выведет print(3 // 2) ?

1) 1.5
2) 1
3) 2
4) Error

Раздел 3. Классы, интерфейсы, пакеты

Контролируемые ИДК: ОПК-3.1

Тема 3.1. Классы, интерфейсы, пакеты

Форма контроля/оценочное средство: Тестирование

Вопросы/Задания:

1. Какая конструкция используется для обработки исключений?

1) if-else
2) try-except
3) switch-case
4) while-else

2. Что такое self в методах класса?

1) Ключевое слово
2) Ссылка на экземпляр класса
3) Глобальная переменная
4) Имя функции

3. Какой метод вызывается при создании объекта класса?

1) create
2) new
3) init
4) start

4. Как открыть файл для чтения в Python?

1) file = open("file.txt", "r")
2) file = read("file.txt")
3) file = open("file.txt", "read")
4) file = file.open("file.txt")

5. Что выведет следующий код?

```
def func(x=[]):
    x.append(1)
    return x
print(len(func()))
print(len(func()))
```

1) 1, 1
2) 1, 2
3) 0, 1
4) Error

Раздел 4. Обработка исключений

Контролируемые ИДК: ОПК-3.1

Тема 4.1. Обработка исключений

Форма контроля/оценочное средство: Тестирование

Вопросы/Задания:

1. Какая функция преобразует список в итератор?

1) list()
2) iter()
3) next()
4) range()

2. Что такое lambda в Python?

1) Ключевое слово цикла
2) Анонимная функция
3) Тип данных
4) Оператор

3. Какой модуль используется для работы с регулярными выражениями?

1) regex
2) re
3) regexp
4) pattern

4. Что выведет print([1, 2, 3] + [4, 5]) ?

1) [1, 2, 3, 4, 5]
2) [5, 7, 8]
3) Error
4) [1, 2, 3], [4, 5]

5. Как создать пустой словарь?

1) dict = []
2) dict = {}
3) dict = ()
4) dict = set()

Раздел 5. Ввод-вывод

Контролируемые ИДК: ОПК-3.1

Тема 5.1. Ввод-вывод

Форма контроля/оценочное средство: Тестирование

Вопросы/Задания:

1. Что такое GIL в Python?

1) Графический интерфейс
2) Global Interpreter Lock
3) Garbage Collector
4) Generic Iterator Library

2. Какой декоратор делает метод статическим?

1) @static
2) @staticmethod
3) @classmethod
4) @property

3. Что выведет следующий код?

```
class A:
    x = 1
a1 = A()
a2 = A()
A.x = 2
print(a1.x, a2.x)
```

1) 1, 1

2) 2, 2
3) 1, 2
4) 2, 1

4. Какой модуль используется для создания виртуального окружения?

1) virtualenv
2) venv
3) pyenv
4) env

5. Что такое comprehension в Python?

1) Способ создания коллекций
2) Метод сжатия
3) Тип данных
4) Функция

Раздел 6. Параметризация типов в Python

Контролируемые ИДК: ОПК-3.1

Тема 6.1. Параметризация типов в Python

Форма контроля/оценочное средство: Тестирование

Вопросы/Задания:

1. Как правильно импортировать конкретную функцию из модуля?

1) import module.function
2) from module import function
3) import function from module
4) using module import function

2. Что выведет `print(2 3 2)`?

1) 64
2) 512
3) 18
4) Error

3. Какой протокол должен реализовывать объект для использования в цикле for?

1) Iterable
2) Callable
3) Hashable
4) Comparable

4. Что такое monkey patching?

1) Исправление ошибок
2) Динамическое изменение кода во время выполнения
3) Тестирование
4) Отладка

5. Какой магический метод отвечает за преобразование объекта в строку?

1) string
2) str
3) text
4) format

8. Оценочные материалы промежуточной аттестации

3 семестр, Зачет с оценкой

Контролируемые ИДК: ОПК-3.1

Вопросы/Задания:

1. Дайте определение алгоритма. Перечислите основные свойства алгоритмов.

2. Какие существуют виды алгоритмов по способу представления? (Способы описания алгоритмов).

3. Что такое блок-схема? Опишите основные элементы блок-схемы и их назначение.

4. Перечислите и охарактеризуйте основные алгоритмические структуры.

5. Опишите алгоритм разветвляющейся структуры. Приведите примеры.

6. Опишите алгоритм циклической структуры. Приведите примеры.

7. В чем особенность алгоритма цикла с предусловием? Приведите пример.

8. В чем особенность алгоритма цикла с постусловием? Приведите пример.

9. В чем особенность алгоритма цикла со счетчиком? Приведите пример.

10. Какие принципы работы с интерпретатором Python вы знаете? (Чем интерпретатор отличается от компилятора?).

11. Опишите базовую структуру программы на Python.

12. Какие основные типы данных существуют в Python? Приведите примеры.

13. Организация ввода и вывода на экран в Python. Функции input() и print().

14. Математические операции в Python. Приоритет операций. Приведите примеры.

15. Ветвления в Python. Операторы if, elif, else. Пример простого ветвления.

16. Множественное ветвление в Python. Как оно реализуется?

17. Циклы в Python. Виды циклов (while, for). Их синтаксис и примеры использования.

18. Операторы управления циклами (break, continue, pass). Их назначение и примеры использования.

19. Тип данных "список" (list) в Python. Методы, функции (len(), sum(), min(), max()) и операции (+, *, in) для работы со списками. Основные методы (добавление, удаление, поиск).

20. Тип данных "кортеж" (tuple) в Python. Чем он отличается от списка? Методы, функции и операции для работы с кортежами. Неизменяемость.

21. Тип данных "строка" (str) в Python. Методы, функции и операции для работы со строками. Приведите примеры основных методов (поиск, замена, разделение, форматирование).

22. Создание подпрограмм (функций) в Python. Ключевое слово def. Структура функции.

23. Определение пользовательских функций в Python. Приведите пример.

24. Способы передачи параметров в функции Python (позиционные, ключевые, значения по умолчанию).

25. Возврат значений из функции. Оператор return. Может ли функция возвращать несколько значений? Как?

26. Что такое класс в Python? Как описывается класс (class)? Что такое объект (экземпляр класса)?

27. Что такое конструктор объекта в Python? Какой специальный метод его реализует (__init__)?

28. Что такое статические элементы класса (переменные, методы)? Ключевое слово @staticmethod. Чем они отличаются от обычных методов?

29. Что такое деструктор объекта? Какой специальный метод его реализует (__del__)? Когда он вызывается?

30. Что такое наследование в ООП? Какие формы наследования существуют (одиночное, множественное)? Как реализуется наследование в Python?

31. Что такое инкапсуляция и полиморфизм в контексте ООП?

32. Что такое обработка исключительных ситуаций (исключений)? Конструкции try-except-else-finally в Python. Пример использования.

33. Что такое интерфейс в контексте ООП? (Хотя в Python это понятие реализуется иначе, чем в строго типизированных языках, вопрос уместен). Как можно эмулировать интерфейсы в Python?

34. Файловый ввод-вывод в Python. Функции open(), close(). Основные режимы открытия файла ('r', 'w', 'a', 'b', '+').

35. Основные методы для чтения (read(), readline(), readlines()) и записи (write(), writelines()) в файл.

36. Что такое итератор файла? Как осуществляется чтение строк с помощью итераторов файлов (например, в цикле for line in file_object)? В чем преимущество такого подхода?

37. Что подразумевается под "параметризацией типов" в Python?

38. Что такое "формы вложения"? Приведите пример вложенного класса.

9. Материально-техническое и учебно-методическое обеспечение дисциплины

9.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Советов Б. Я. Информационные технологии: теоретические основы: учебное пособие для вузов / Б. Я. Советов, В. В. Цехановский. - 3-е изд., стер. - Санкт-Петербург: Лань, 2022. - 444 с. - 978-5-507-45305-4. - Текст: электронный // Издательство Лань: [сайт]. - URL: https://e.lanbook.com/book/264935 - Режим доступа: по подписке

Дополнительная литература

1. Цветкова А. В. Информатика и информационные технологии: учебное пособие / А. В. Цветкова. - Информатика и информационные технологии - Саратов: Научная книга, 2012. - 189 с. - 2227-8397. - Текст: электронный // IPR SMART: [сайт]. - URL: https://www.iprbookshop.ru/6276.html - Режим доступа: по подписке

9.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся

Профессиональные базы данных

1. https://student2.consultant.ru/cgi/online.cgi - Правовой Сервер КонсультантПлюс

Ресурсы «Интернет»

1. https://intuit.ru/ - Электронная образовательная среда "ИНТУИТ"
2. https://znanium.ru/ - Электронно-библиотечная система "Znanium"
3. https://www.elibrary.ru/ - НЭБ eLIBRARY
4. https://rusneb.ru/ - Национальная электронная библиотека (НЭБ)

9.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения
(обновление производится по мере появления новых версий программы)

1. Sublime Text 3;
2. Microsoft Visual Studio;

Перечень информационно-справочных систем
(обновление выполняется еженедельно)

Не используется.

9.4. Оборудование и технические средства обучения

Компьютерный класс

Маркерная доска - 1 шт.
Моноблок - 96 шт.
Проектор - 7 шт.
Автоматизированное рабочее место (АРМ) - 15 шт.
Плоттер - 1 шт.
Сканер - 1 шт.

10. Методические указания по освоению дисциплины (модуля)

С целью успешного освоения данного предмета студенту необходимо выполнять домашние работы по указанным преподавателям темам и представленным на занятиях материалам. Для более успешного выполнения работы студенту предлагается использование учебно-методических материалов, соответствующих указанной теме.

Самостоятельная работа студентов описывается и регулируется методическими рекомендациями по дисциплине.

Описание возможностей изучения дисциплины лицами с ОВЗ и инвалидами

При необходимости обучения студентов-инвалидов и лиц с ограниченными возможностями здоровья аудиторные занятия могут быть заменены или дополнены изучением полнотекстовых лекций, презентаций, видео- и аудиоматериалов в электронной информационно-образовательной среде (ЭИОС) университета. Индивидуальные задания подбираются в адаптированных к ограничениям здоровья формах (письменно или устно, в форме презентаций). Выбор методов обучения зависит от их доступности для инвалидов и лиц с ограниченными возможностями здоровья.

Рабочая программа дисциплины для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.