


ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО РЫБОЛОВСТВУ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КамчатГТУ»)

Факультет информационных технологий, экономики и управления

Кафедра «Системы управления»

УТВЕРЖДАЮ
Декан ФИТЭУ

 /И.А. Рычка/

«28» января 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Основы программирования python»

направление подготовки:
27.03.04 «Управление в технических системах»
(уровень бакалавриата)

направленность (профиль): «Управление и информатика в технических системах»

Рабочая программа дисциплины разработана в соответствии с ФГОС ВО по направлению подготовки 27.03.04 «Управление в технических системах» студентов очной и заочной форм обучения, профиль «Управление и информатика в технических системах» и учебного плана ФГБОУ ВО «КамчатГТУ».

Составитель рабочей программы:

старший преподаватель кафедры СУ

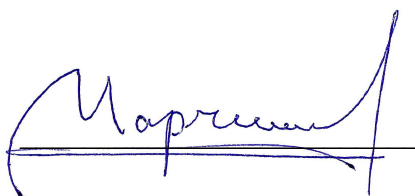


Е.А. Лутцева

Рабочая программа рассмотрена на заседании кафедры «Системы управления»

Протокол № 5 от «20» декабря 2025 года.

«20» декабря 2025 г.



Заведующий кафедрой
«Системы управления»
А.А. Марченко

1. ЦЕЛИ И ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целью освоения дисциплины является изучение базовых понятий и терминов программирования как науки, изучение конструкции языка программирования Python, развитие навыков программирования на языке Python.

Задачи освоения дисциплины:

- изучение основ программирования на языке Python;
- изучение базовых конструкций и нетривиальных возможностей языка программирования Python;
- изучение возможностей различных пакетов языка Python.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

В результате изучения дисциплины у студента должны быть сформированы следующие общепрофессиональные компетенции:

– способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности (ОПК-6).

Наименование компетенции при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы, представлены в таблице 1.

Таблица 1 – Планируемые результаты обучения по дисциплине, соотнесенные с установленными в программе бакалавриата индикаторами достижения компетенций

Код компетенции	Наименование компетенции	Код и наименование индикатора достижения ПК	Планируемый результат обучения по дисциплине	Код показателя освоения
ОПК-6	способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии	ИД-1 _{ОПК-6} Знает основы написания программного кода для работы систем автоматического управления	Знать: – синтаксис и базовые конструкции языка программирования python; – библиотеки (пакеты) python для научных вычислений.	З(ОПК-6)1 З(ОПК-6)2
		ИД-1 _{ОПК-6} Умеет разрабатывать и использовать алгоритмы и программы для функционирования систем автоматического управления	Уметь: – реализовывать алгоритмы для проведения и обработки результатов вычислительных экспериментов с использованием языка программирования python; – использовать библиотеки (пакеты) python для научных вычислений.	У(ОПК-6)1
		ИД-3 _{ОПК-6} Владеет навыками работы в системах, связанных с современными информационными технологиями	Владеть: – методами разработки программ для проведения и обработки результатов эксперимента на языке python ; – инструментами для написания программ, использующих научные вычисления. – практическими навыками	У(ОПК-6)2
				В(ОПК-6)1 В(ОПК-6)2

			разработки и реализации алгоритмов на базе языков и пакетов прикладных программ моделирования.	В(ОПК-6)3
--	--	--	------------------------------------------------------------------------------------------------	-----------

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Курс «Основы программирования python» ориентирован на подготовку бакалавров по направлению 27.03.04 «Управление в технических системах». Данная дисциплина относится к блоку Б1.О – дисциплины обязательной части.

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Тематический план дисциплины для студентов очной формы обучения

Наименование разделов и тем	Всего часов	Контактная работа	Контактная работа по видам учебных занятий				Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			Лекции	Практические занятия	Лабораторные работы	СРП			
1	2	3	4	5	6	7	8	9	10
Тема 1. Обзор языков программирования. Технологии разработки программ	12	6	8	0	0	0	6	Контроль СРС, защита лабораторных работ, вопросы, выносимые на рассмотрение	
Тема 2. Введение в python	22	14	6	0	6	0	8		
Тема 3. Типы данных	26	16	8	0	10	0	10		
Тема 4. Специфические возможности python. Обработка данных	46	30	12	0	18	0	16		
Экзамен								Опрос	36
Всего	144	68	34	0	34	0	40		36

4.2. Тематический план дисциплины для студентов заочной формы обучения

Наименование разделов и тем	Всего часов	Контактная работа	Контактная работа по видам учебных занятий				Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			Лекции	Практические занятия	Лабораторные работы	СРП			
1	2	3	4	5	6	7	8	9	10
Тема 1. Обзор языков программирования. Технологии разработки программ	20	0	0	0	0	0	20	Контроль СРС, защита лабораторных работ,	
Тема 2. Введение в python	33	6	2	0	4	0	27		

Тема 3. Типы данных	36	6	2	0	4	0	30	вопросы, выносимые на рассмотрение	
Тема 4. Специфические возможности python. Обработка данных	46	6	2	0	4	0	40		
Экзамен								Опрос	9
Всего	144	18	6	0	12	0	117		9

4.3. Содержание дисциплины

Тема 1. Обзор языков программирования. Технологии разработки программ.

Лекция

Понятие языка программирования. Развитие языков. Классификация языков: машинный язык, низкоуровневый язык, высокоуровневый язык. Понятие парадигмы программирования. Обзор парадигм программирования. Понятие трансляции. Трансляция и интерпретация. Виды трансляторов. Технология разработки программного обеспечения. Жизненный цикл программного продукта. Модели жизненного цикла.

Основные понятия темы: язык программирования, парадигма программирования, компилятор, интерпретатор, жизненный цикл ПО.

Самостоятельная работа студента

Изучение дополнительного теоретического материала, подготовка доклада.

Литература: [5]

Тема 2. Введение в python

Лекция

История создания Python. Философия Python. Функции, осуществляющие ввод-вывод на консоль. Использование различных математических операций в Python. Подключение пакетов и импорт функций. Первое консольное приложение. Ветвящиеся и циклические алгоритмы. Реализация полного, неполного и множественного ветвлений. Различные виды циклов. Операторы управления циклом. Циклы и else.

Основные понятия темы: ввод/вывод, ветвление, цикл.

Лабораторные работы

Лабораторная работа № 1. Знакомство с Python. Ввод-вывод. Математические функции.

Лабораторная работа № 2. Условные операторы.

Лабораторная работа № 3. Циклы.

Самостоятельная работа студента

Изучение литературы, подготовка теоретического материала и данных для выполнения лабораторных работ.

Литература: [1], [2], [3], [4]

Тема 3. Типы данных

Лекция

Тип данных Список. Изменяемый тип данных list. Операторы, функции и методы работы с list. Использование list как массива. Тип данных Кортеж. Изменяемый тип данных tuple. Операторы, функции и методы работы с tuple. Отличия tuple и list. Тип данных Множество. Неизменяемый тип set. Операторы, функции и методы работы с set. Операции пересечения, объединения, разности и дополнения множеств. Тип данных Словарь. Изменяемый тип данных dict. Пары «ключ-значение». Операторы, функции и методы работы с dict. Строки. Тип данных Строка. Методы, функции и операции работы со строками.

Основные понятия темы: коллекции данных: кортежи, списки, строки, множества, словари.

Лабораторные работы

Лабораторная работа № 4. Списки и кортежи.

Лабораторная работа № 5. Строки.

Лабораторная работа № 6. Множества.

Лабораторная работа № 7. Словари.

Самостоятельная работа студента

Изучение литературы, подготовка теоретического материала и данных для выполнения лабораторных работ.

Литература: [1], [2], [3], [4]

Тема 4. Специфические возможности python. Обработка данных

Лекция

Внешние подпрограммы. Функции и процедуры. Реализация функций и процедур в Python. Формальные и фактические параметры функций. Параметры со значениями по умолчанию. Функции с переменным числом параметров. Написание программы использующей внешние функции. Работа с текстовыми и бинарными файлами. Определение файла. Отличия текстового и бинарного файлов. Запись и чтение из текстового файла средствами Python. Запись и чтение из бинарного файла средствами Python. Обработка исключений в Python. Определения исключения. Виды исключений в Python. Механизмы генерации, ловли и обработки исключений в Python. Регулярные выражения в Python. Основы создания и обработки регулярных выражений с использованием Python. Обработка данных в python: подготовка и исследование данных, работа с данными и их визуализация. Библиотеки обработки данных: numpy, pandas. Библиотеки визуализации: matplotlib, seaborn.

Основные понятия темы: функции, процедуры, файлы, исключения, обработка данных (NumPy, seaborn).

Лабораторные занятия

Лабораторная работа № 8. Функции и процедуры.

Лабораторная работа № 9. Работа с файлами.

Лабораторная работа № 10. Регулярные выражения. Библиотека RE.

Лабораторная работа № 11. Исключения в python.

Лабораторная работа № 12. Библиотека NumPy.

Лабораторная работа № 13. Основы работы с данными. Подготовка и исследование данных, работа с данными и их визуализация.

Самостоятельная работа студента

Изучение литературы, подготовка теоретического материала и данных для выполнения лабораторных работ, подготовка к экзамену.

Литература: [1], [2], [3], [4]

5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Самостоятельная работа студентов по дисциплине «Основы программирования python» является важной составляющей частью подготовки студентов по направлению подготовки 27.03.04 «Управление в технических системах» и выполняется в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) и учебным планом КамчатГТУ.

Самостоятельная работа студентов ставит своей целью:

1. Развитие навыков ведения самостоятельной работы;
2. Приобретение опыта систематизации полученных результатов исследований, формулировку новых выводов и предложений как результатов выполнения работы;
3. Развитие умения использовать научно-техническую литературу и нормативно-методические материалы в практической деятельности;
4. Приобретение опыта публичной защиты результатов самостоятельной работы.

Внеаудиторная самостоятельная работа при изучении курса включает в себя следующие виды работ:

- проработка (изучение) материалов лекций;
- чтение и проработка рекомендованной основной и дополнительной литературы;
- подготовка к лабораторным работам;

- поиск и проработка материалов из Интернет-ресурсов, периодической печати;
- подготовка к текущему и итоговому (промежуточная аттестация) контролю знаний по дисциплине.
- подготовка к текущему и итоговому контролю знаний по дисциплине.

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Основы программирования python» представлен в приложении к рабочей программе дисциплины и включает в себя:

- перечень компетенций, с указанием этапов их формирования в процессе освоения образовательной программы;
- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- типовые контрольные задания или материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
- методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Перечень вопросов для проведения промежуточной аттестации по дисциплине (экзамен)

1. Встроенный тип str. Методы объекта str.
2. print() и форматирование вывода, считывание данных с помощью input().
3. Работа с файловой системой средствами Python.
4. Работа с файлами. Методы open(), close(), read(), write().
5. Unicode- строки, типы bytes и bytearray.
6. Встроенные типы последовательностей list, tuple, range и их методы.
7. Встроенный объект dict и его методы.
8. Встроенные типы чисел – int, float, complex. Машинное представление чисел с плавающей точкой и целых. Преобразование типов при сравнении чисел.
9. Двоичное представление чисел. Неассоциативность операций в арифметике с плавающей запятой. Целые числа с произвольной точностью.
10. Множества. Встроенные типы set и frozenset.
11. Инструкции if/else/elif, логические операторы и выражения сравнения.
12. Циклы while и for в Python.
13. Функции в Python. Основные понятия.
14. Области видимости и пространство имен в Python.
15. Передача аргументов в функцию. Специальные режимы сопоставления аргументов.
16. Парадигма объектно-ориентированного программирования. Поддержка в Python функционального программирования.
17. Объекты. Динамическая типизация. Инкапсуляция.
18. Генерация объекта class. Новое пространство имен. Объект экземпляр класса.
19. Атрибуты класса. Атрибуты данных. Атрибуты-методы. Параметр self.
20. Специальные методы и атрибуты классов. Методы __init__() и __del__() в Python.
21. Абстрактные методы в Python. Классические классы и классы нового стиля.
22. Наследование. Базовый и производный класс. Построение производного класса.
23. Полиморфизм. Подмена методов в производном классе. Доступ к методам базового класса.

24. Обработка исключений. Инструкция try... except... else... finally. Классы встроенных исключений. Пользовательские исключения. Генерация заданного исключения. Поиск ошибок программирования на стадиях разработки и тестирования.
25. Генераторы Python.
26. Модули и пакеты. Библиотеки сторонних разработчиков (Django, Flask, Pygame, PyQt, twisted).
27. Модули и пакеты. Графический интерфейс.
28. Пакет NumPy.
29. Пакет SciPy.
30. Пакет Matplotlib.

Перечень практических заданий на экзамен

1. Написать программу определения минимального из трех чисел.
2. Написать программу определения размера стипендии. Пользователь вводит 3 натуральных числа: общее количество оценок, количество пятерок, количество четверок. Если все пятерки – стипендия 6000 руб., если одна четверка – 4500 руб., если две четверки – 3750 руб., если нет троек – 3000 руб., иначе – нет стипендии.
3. Написать программу определения четности/нечетности числа.
4. Создать список, состоящий из степеней двойки от 0й до 10й. 1 2 4 8 16 ... 1024. Двумя способами - с помощью цикла и с помощью генератора.
5. Создать список, состоящий из квадратов натуральных чисел от 1 до 10. 1 4 9 ... 10000. Двумя способами - с помощью цикла и с помощью генератора.
6. Подсчитать количество уникальных чисел в последовательности, вводимой пользователем.
7. Вывести слова, входящие в состав предложения, без повторений.
8. Написать процедуру, здоровающуюся с пользователем. Имя пользователя вводится как параметр процедуры.
9. Написать функцию определения корней квадратного уравнения. В качестве параметров подаются коэффициенты a, b, c ($ax^2 + bx + c = 0$).
10. Написать программу, разбивающую строку на отдельные слова. Вывести слова в алфавитном порядке.
11. Написать программу, осуществляющую замену подстроки в строке. Например, дана строка: “Привет! Меня зовут Ольга”, произвести замену “Ольга -> Анна”.
12. Написать программу, вычисляющую пересечение и объединение двух множеств. Элементы множества вводятся пользователем с клавиатуры.
13. Создать словарь, состоящий из пар «страна – столица» (РФ-Москва, США-Вашингтон, Белоруссия-Минск, Украина-Киев, Китай-Пекин).
14. Написать программу, обрабатывающую исключение – деление на 0.
15. Написать программу, реализующую класс с конструктором. Создать несколько объектов данного класса.

7. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

7.1. Основная литература

1. Программирование на языке высокого уровня Python : Учебное пособие для вузов / Федоров Д. Ю. - 3-е изд. ; пер. и доп. - Москва : Юрайт, 2021. - 210 с. - (Высшее образование). - Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользова-

телей. - URL: <https://urait.ru/bcode/478098> (дата обращения: 11.01.2022). - ISBN 978-5-534-14638-7

2. Основы программирования на Python : Учебное пособие для вузов / Чернышев С. А. - Москва : Юрайт, 2021. - 286 с. - (Высшее образование). - Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей. - URL: <https://urait.ru/bcode/477353> (дата обращения: 11.01.2022). - ISBN 978-5-534-14350-8
3. Программирование в Python 3 [Электронный ресурс] : учебное пособие / Полупанов Д. В., Абдюшева С. Р., Ефимов А. М. - Уфа : БашГУ, 2020. - 164 с. - ISBN 978-5-7477-5230-6.

7.2. *Дополнительная литература*

4. Программирование на языке Python [Электронный ресурс] : учебное пособие для лабораторных занятий / Широбокова С. Н., Кацупеев А. А., Сулыз А. В. - Новочеркасск : ЮРГ-ПУ, 2020. - 104 с. - ISBN 978-5-9997-0725-3.
5. Технологии и методы программирования : Учебное пособие для вузов / Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю. - Москва : Юрайт, 2021. - 235 с. - (Высшее образование). - Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей. - URL: <https://urait.ru/bcode/469759> (дата обращения: 11.01.2022). - ISBN 978-5-534-02816-4

8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ»

1. Электронно-библиотечная система «eLibrary»: [Электронный ресурс]. – Режим доступа: <http://www.elibrary.ru>.
2. Электронная библиотечная система «IPRbooks» [Электронный ресурс]. – Электронные данные – Режим доступа : <http://www.iprbookshop.ru/>.
3. Электронная информационная образовательная среда LMS Moodle [Электронный ресурс]. – Режим доступа: <https://lk.kstu.su>.

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методика преподавания данной дисциплины предполагает чтение лекций, проведение практических и лабораторных занятий, групповых и индивидуальных консультаций по отдельным (наиболее сложным) специфическим проблемам дисциплины. Предусмотрена самостоятельная работа студентов, прохождение аттестационных испытаний промежуточной аттестации (экзамен).

Лекции проводятся, как правило, в интерактивной форме с элементами дискуссий, и спорных посылов и утверждений. На лекциях преподаватель знакомит слушателей с основными понятиями и положениями по текущей теме. При проведении лекций используются современные информационные технологии, демонстрационные материалы

Лабораторная работа – это выполнение студентами под руководством преподавателя или по инструкции заданий с применением персонального компьютера.

Лабораторные работы составляют важную часть теоретической и профессиональной практической подготовки обучающихся. Они направлены на формирование учебных и профессиональных практических умений. Лабораторные занятия носят систематический характер, регулярно следуя за лекционными занятиями. Лабораторные работы выполняются согласно графику, при этом соблюдается принцип индивидуального выполнения работ (в некоторых случаях – группового).

Проведение лабораторных работ включает в себя следующие этапы:

- постановку темы занятия и определение задач лабораторной работы;

- определение порядка лабораторной работы или отдельных ее этапов;
- непосредственное выполнение лабораторной работы студентами с соблюдением техники безопасности;
- подведение итогов лабораторной работы и формулирование основных выводов.

При подготовке к занятию необходимо обратить внимание на цель занятия, на основные вопросы для подготовки к занятию, на содержание темы занятия. Этапы подготовки к практическому занятию:

- освежить в памяти теоретические сведения, полученные на лекциях и в процессе самостоятельной работы,
- подобрать необходимую учебную и справочную литературу.

В течение лабораторной работы студенту необходимо выполнить индивидуальные или групповые задания, выданные преподавателем, а затем оформить получившиеся результаты в виде отчёта, который выполняется в соответствии с нижеизложенными указаниями по оформлению письменных отчётов. Помимо этого, студенту необходимо подготовить ответы на примерный перечень вопросов по теме работы.

10. КУРСОВОЙ ПРОЕКТ (РАБОТА)

В соответствии с учебным планом курсовой проект (работа) по дисциплине «Основы программирования python» не предусмотрен.

11. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

11.1 Перечень информационных технологий, используемых при осуществлении образовательного процесса:

- электронные образовательные ресурсы, представленные в п.8 рабочей программы;
- использование слайд-презентаций;
- интерактивное общение с обучающимися и консультирование в электронной информационной образовательной среде ФГБОУ ВО «КамчатГТУ».

11.2 Перечень программного обеспечения, используемого при осуществлении образовательного процесса:

- операционные системы Astra Linux (или иная операционная система, включенная в реестр отечественного программного обеспечения);
- комплект офисных программ Р-7 Офис (в составе текстового процессора, программы работы с электронными таблицами, программные средства редактирования и демонстрации презентаций);
- программа проверки текстов на предмет заимствования «Антиплагиат»;
- среда разработки PyCharm;
- язык python 3.x.

11.3 Перечень информационно-справочных систем:

- справочно-правовая система «Гарант»;
- портал Федеральных государственных образовательных стандартов высшего образования (<https://fgosvo.ru>).

12. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

- для проведения занятий лекционного типа, практических (семинарских) занятий, лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и

промежуточной аттестации; для самостоятельной работы обучающихся – учебная аудитория № 7-510 («Лаборатория сетевых технологий», «Кабинет самостоятельной работы студентов»), оборудованная 10 рабочими станциями с доступом к сети «Интернет» и к электронной информационной образовательной среде, с комплектом учебной мебели на 12 посадочных мест;

- доска аудиторная;
- интерактивная доска;
- презентации по темам курса «Основы программирования python».