

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Костромской государственный университет»
(КГУ)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Основы программирования на Python

Направление 29.03.02 Технологии и проектирование текстильных изделий

**Профиль Технологии цифрового проектирования композиционных
материалов**

Квалификация выпускника: Бакалавр

**Кострома
2025**

Рабочая программа дисциплины «Основы программирования на Python» разработана в соответствии с Федеральным государственным образовательным стандартом по направлению подготовки 29.03.02 Технологии и проектирование текстильных изделий" № 963 от 22.09.2017

Разработал:

Сокова Галина Георгиевна, профессор кафедры ТММ, ДМ и ПТМ

Рецензент: Гречухин Александр Павлович, профессор кафедры Технологии и проектирования тканей и трикотажа ФГБОУ ВО «Костромской государственной университет», д.т.н.

ПРОГРАММА УТВЕРЖДЕНА:

Заведующий кафедрой Технологии и проектирования тканей и трикотажа:

Корабельников А.Р., д.т.н., профессор

Протокол заседания кафедры № 8 от 30.06.2025 г.

1. Цели и задачи освоения дисциплины

Цель дисциплины:

сформировать у студентов базовые навыки программирования на языке Python, необходимые для решения инженерных задач, связанных с математическим анализом, моделированием и автоматизацией профессиональной деятельности.

Задачи дисциплины

1. Ознакомить студентов с основами синтаксиса и семантики языка Python.
2. Научить разрабатывать алгоритмы и реализовывать их на Python для решения инженерных задач.
3. Познакомить с основами работы с данными, визуализацией и математическими библиотеками (NumPy, Matplotlib).
4. Развить навыки самостоятельного поиска и использования информационных ресурсов для программирования.
5. Обеспечить умение применять программные средства для анализа и моделирования инженерных процессов.

2. Перечень планируемых результатов обучения по дисциплине

В результате освоения дисциплины обучающийся должен:

освоить компетенции:

ОПК-1 Способен решать вопросы профессиональной деятельности на основе естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования

знать:

1. Основы программирования:
 - Понимание принципов алгоритмизации и логического мышления.
 - Знание синтаксиса и семантики языка Python.
 - Понимание основных структур данных (списки, кортежи, словари, множества).
2. Программные средства:
 - Знание о библиотеке NumPy для научных вычислений.
 - Понимание основ визуализации данных с помощью Matplotlib.
 - Знание методов работы с файлами (чтение, запись, обработка).
3. Математические и инженерные аспекты:
 - Основы математического анализа и моделирования.
 - Понимание применения программирования для решения инженерных задач.

Уметь:

1. Разработка алгоритмов:
 - Умение разрабатывать и реализовывать алгоритмы на Python для решения практических задач.
 - Способность применять условные конструкции и циклы для управления потоком выполнения программы.
2. Работа с данными:
 - Умение обрабатывать и анализировать данные с использованием Python.
 - Способность визуализировать результаты анализа с помощью графиков и диаграмм.
3. Автоматизация процессов:
 - Умение автоматизировать рутинные задачи с помощью написания скриптов.
 - Способность анализировать и интерпретировать результаты, полученные в ходе

программирования.

4. Самостоятельное обучение:

- Умение самостоятельно искать и использовать ресурсы для изучения новых библиотек и инструментов.
- Способность адаптировать полученные знания и навыки к конкретным инженерным задачам.

3. Место дисциплины в структуре ОП ВО

Дисциплина относится блоку Б1 базовой части учебного плана (Б1.О.32) Изучается в 3 семестре.

Дисциплина " Основы программирования на Python " является базовой для успешного освоения общенаучных и специальных дисциплин (профессиональный цикл), изучающих конкретные задачи, таких как "Химия", "Инженерная и компьютерная графика", "Физика", "Теория вероятности и математическая статистика в профессиональной деятельности», «Прикладная механика» и др.

4. Объем дисциплины 4.1. Объем дисциплины в зачетных единицах с указанием академических часов и виды учебной работы

Виды учебной работы	Очная форма
	Всего
Общая трудоемкость в зачетных единицах	3
Общая трудоемкость в часах	108
Аудиторные занятия в часах, в том числе:	50
Лекции	16
Практические занятия	-
Лабораторные занятия	34
Самостоятельная работа, в часах	58
Контроль	-
Форма промежуточной аттестации	зачет
ИКР	-

4.2 Объем контактной работы на 1 обучающегося

Виды учебных занятий	Очная форма
	Всего
Лекции	16
Практические занятия	-
Лабораторные занятия	34
Консультации	-
Зачет/зачеты	-
Экзамен/экзамены	-
Курсовые работы	-
Курсовые проекты	-
Всего	50

5. Содержание дисциплины, структурированное по темам (разделам), с указанием количества часов и видов занятий

5.1 Тематический план учебной дисциплины

№	Название раздела, темы	Всего час	Аудиторные		Самостоятельная работа
			Лекц.	лабораторные	
1	Введение в программирование и Python. Установка среды разработки	2	2		
	Установка и настройка Python и среды разработки	6		2	4
	Решение простых задач: ввод, вывод, арифметика	6		2	4
2	Основы синтаксиса Python: переменные, типы данных, ввод/вывод	2	2		
	Использование условных операторов и циклов	6		2	4
	Создание и вызов функций	6		2	4
3	Управляющие конструкции: условные операторы, циклы	2	2		
	Работа со списками и словарями	6		2	4
	Обработка текстовых файлов	6		2	4
4	Функции и модульность программ. Аргументы и возвращаемые значения	2	2		
	Решение инженерных задач с использованием функций	6		2	4
5	Структуры данных: списки, кортежи, словари, множества	2	2		
	Введение в NumPy: массивы и операции над ними	12		4	8
6	Работа с файлами: чтение и запись данных	2	2		
	Применение NumPy для решения задач математического анализа.	12		4	8
7	Введение в библиотеки для научных вычислений: NumPy	2	2		
	Проектная работа: моделирование простого инженерного процесса	14		6	8
8	Визуализация данных	2	2		
	Визуализация данных: построение графиков	6		4	2
	Подготовка к зачету	6			6
	Всего	108	16	34	58

5.2. Содержание

1. Введение в программирование и Python. Установка среды разработки
 - Что такое программирование и зачем оно нужно инженеру-технологу.
 - Обзор языка Python: история, особенности, сфера применения.
 - Установка и настройка среды разработки (например, PyCharm, VS Code, Jupyter Notebook).
 - Запуск первой программы «Hello, World!».
2. Основы синтаксиса Python: переменные, типы данных, ввод/вывод
 - Переменные и правила их именования.
 - Основные типы данных: целые числа, числа с плавающей точкой, строки, булевы значения.
 - Операции с данными.
 - Ввод данных с клавиатуры и вывод на экран.
3. Управляющие конструкции: условные операторы, циклы
 - Логические выражения и операторы сравнения.
 - Условные конструкции: if, elif, else.

- Циклы for и while, их применение.
 - Управляющие операторы break и continue.
4. Функции и модульность программ. Аргументы и возвращаемые значения

- Определение функций, их назначение.
 - Параметры и аргументы функций.
 - Возврат значений из функции.
 - Область видимости переменных.
 - Модули и импорт библиотек.
5. Структуры данных: списки, кортежи, словари, множества
- Списки: создание, индексация, методы.
 - Кортежи: особенности и отличия от списков.
 - Словари: ключи и значения, основные операции.
 - Множества: уникальные элементы, операции с множествами.
6. Работа с файлами: чтение и запись данных
- Открытие и закрытие файлов.
 - Чтение данных из файлов: построчно и целиком.
 - Запись данных в файлы.
 - Работа с текстовыми и бинарными файлами.
 - Обработка исключений при работе с файлами.
7. Введение в библиотеки для научных вычислений: NumPy
- Знакомство с библиотекой NumPy.
 - Создание и манипуляции с массивами.
 - Основные математические операции над массивами.
 - Применение NumPy для обработки инженерных данных.
8. Визуализация данных с помощью Matplotlib
- Основы построения графиков.
 - Создание линейных графиков, гистограмм, точечных диаграмм.
 - Настройка внешнего вида графиков: подписи, легенды, оси.
 - Сохранение графиков в файл.

6. Методические материалы для обучающихся по освоению дисциплины

6.1. Самостоятельная работа обучающихся по дисциплине

№	Название раздела, темы	Задание	Всего час	Методические рекомендации	Форма контроля
1	Введение в программирование и Python. Установка среды разработки		2		
	Установка и настройка Python и среды разработки	<ul style="list-style-type: none"> - Установить Python и выбранную среду разработки (PyCharm, VS Code или Jupyter Notebook). - Создать и запустить программу, выводящую на экран приветствие. - Сделать скриншот рабочего окна с запущенной программой. 	6	Изучение материалов лекций и рекомендаций моей литературы	Контроль выполнения задания
	Решение простых задач: ввод, вывод, арифметика	<ul style="list-style-type: none"> - Написать программу, которая запрашивает у пользователя два числа и выводит их сумму, разность, произведение и частное. - Реализовать расчет площади прямоугольника по введённым длинам 	6	Изучение материалов лекций и рекомендаций моей литературы	Контроль выполнения задания

		сторон.			
2	Основы синтаксиса Python: переменные, типы данных, ввод/вывод		2		
	Использование условных операторов и циклов	<ul style="list-style-type: none"> - Написать программу, которая определяет, является ли введённое число положительным, отрицательным или нулём. - Создать программу для вычисления факториала числа с использованием цикла. - Написать программу, которая выводит все чётные числа от 1 до 50. 	6	Изучение материалов лекций и рекомендации по литературе	Контроль выполнения задания
	Создание и вызов функций	<ul style="list-style-type: none"> - Реализовать функцию для вычисления площади круга по радиусу. - Написать функцию, которая принимает список чисел и возвращает их среднее значение. - Создать программу, которая использует эти функции для решения задачи. 	6	Изучение материалов лекций и рекомендации по литературе	Контроль выполнения задания
3	Управляющие конструкции: условные операторы, циклы		2		
	Работа со списками и словарями	<ul style="list-style-type: none"> Написать программу, которая создаёт список из 10 случайных чисел, находит максимум и минимум. - Реализовать словарь с данными о количестве деталей на складе (название — количество). - Написать программу для обновления и вывода информации из словаря. 	6	Изучение материалов лекций и рекомендации по литературе	Контроль выполнения задания
	Обработка текстовых файлов	<ul style="list-style-type: none"> Создать текстовый файл с данными о технологическом процессе (например, температуры, давления). - Написать программу для чтения данных из файла и вычисления среднего значения. - Сохранить результаты вычислений в новый файл. 	6	Изучение материалов лекций и рекомендации по литературе	Контроль выполнения задания
4	Функции и модульность программ. Аргументы и возвращаемые значения		2		
	Решение инженерных задач с использованием функций	<ul style="list-style-type: none"> Написать программу для расчёта параметров теплового процесса (например, теплопередачи) с использованием функций. - Включить проверку корректности введённых данных. 	6	Изучение материалов лекций и рекомендации по литературе	Контроль выполнения задания
5	Структуры данных: списки, кортежи, словари, множества		2		
	Введение в NumPy: массивы и операции над ними	Создать одномерный и двумерный массивы с	12	Изучение материалов	Контроль выполнения

		помощью NumPy. - Выполнить арифметические операции над массивами. - Написать программу для расчёта среднего и стандартного отклонения массива данных.		лекций и задания рекомендуемой литературы	
6	Работа с файлами: чтение и запись данных		2		
	Применение NumPy для решения задач математического анализа.	Реализовать численное интегрирование функции с использованием массива значений. - Написать программу для решения системы линейных уравнений с помощью NumPy. - Выполнить интерполяцию или аппроксимацию данных.	12	Изучение материалов лекций и задания рекомендуемой литературы	Контроль выполнения задания
7	Введение в библиотеки для научных вычислений: NumPy		2		
	Проектная работа: моделирование простого инженерного процесса	Разработать программу, моделирующую технологический процесс (например, нагрев/охлаждение, движение жидкости). - Использовать функции, массивы NumPy и визуализацию Matplotlib. - Подготовить отчёт с описанием алгоритма, кодом и графиками. - Презентовать результаты работы.	14	Изучение материалов лекций и задания рекомендуемой литературы	Контроль выполнения задания
8	Визуализация данных		2		
	Визуализация данных: построение графиков	Построить линейный график зависимости температуры от времени. - Создать гистограмму распределения значений технологического параметра. - Добавить подписи осей, заголовков и легенду.	6	Изучение материалов лекций и задания рекомендуемой литературы	Контроль выполнения задания
	Подготовка к зачету		6		
	Всего		108		

6.2. Методические рекомендации студентам, изучающим дисциплину «Основы программирования на Python»

Студенту настоятельно рекомендуется посещать лекции ввиду большого объема теоретического и практического материала дисциплины. На лекции нужно обязательно составлять конспект. Это необходимо по той причине, что в виду специфики языка программирования самостоятельная работа с учебной литературой без предварительной подготовки может оказаться весьма затруднительной. За пропущенные лекции и практические занятия студент должен отчитаться перед преподавателем, представив реферат на пропущенную тему и выполнив домашнее задание по теме.

Большое внимание студентам следует уделять самостоятельной работе, которая складывается из изучения материалов лекций и рекомендуемой литературы, подготовке к практическим занятиям

Систематическое изучение материалов лекций и подготовка к практическим занятиям - залог накопления глубоких знаний и успешной сдачи зачета по дисциплине. Студентам следует помнить, что допуском к зачету является освоение учебной программы семестра, что должно быть подтверждено выполненными заданиями. Готовиться к практическим занятиям следует не только теоретически. За период обучения необходимо овладеть навыками практического использования теоретических знаний.

По итогам освоения дисциплины проводится зачет, целью которого является проверка освоенности дисциплины и компетенций.

Зачет преподавателем проводится для студентов, успешно освоивших дисциплину. При возникновении трудностей в изучении того или иного раздела математики студентам рекомендуется посещать консультации преподавателей.

6.2. Тематика и задания для лабораторных работ

1. Установка и настройка Python и среды разработки

- Установить Python и выбранную среду разработки (PyCharm, VS Code или Jupyter Notebook).
- Создать и запустить программу, выводящую на экран приветствие.
- Сделать скриншот рабочего окна с запущенной программой.

2. Решение простых задач: ввод-вывод, арифметика

- Написать программу, которая запрашивает у пользователя два числа и выводит их сумму, разность, произведение и частное.
- Реализовать расчет площади прямоугольника по введенным длинам сторон.

3. Использование условных операторов и циклов

- Написать программу, которая определяет, является ли введенное число положительным, отрицательным или нулем.
- Создать программу для вычисления факториала числа с использованием цикла.
- Написать программу, которая выводит все четные числа от 1 до 50.

4. Создание и вызов функций

- Реализовать функцию для вычисления площади круга по радиусу.
- Написать функцию, которая принимает список чисел и возвращает их среднее значение.
- Создать программу, которая использует эти функции для решения задачи.

5. Работа со списками и словарями (2 ч)

- Написать программу, которая создаёт список из 10 случайных чисел, находит максимум и минимум.
- Реализовать словарь с данными о количестве деталей на складе (название — количество).
- Написать программу для обновления и вывода информации из словаря.

6. Обработка текстовых файлов (2 ч)

- Создать текстовый файл с данными о технологическом процессе (например, температуры, давления).
- Написать программу для чтения данных из файла и вычисления среднего значения.
- Сохранить результаты вычислений в новый файл.

7. Решение инженерных задач с использованием функций (2 ч)

- Написать программу для расчёта параметров теплового процесса (например, теплопередачи) с использованием функций.
- Включить проверку корректности введенных данных.

8. Введение в NumPy: массивы и операции над ними (4 ч)

- Создать одномерный и двумерный массивы с помощью NumPy.
- Выполнить арифметические операции над массивами.
- Написать программу для расчёта среднего и стандартного отклонения массива данных.

9. Применение NumPy для решения задач математического анализа (4 ч)

- Реализовать численное интегрирование функции с использованием массива значений.
- Написать программу для решения системы линейных уравнений с помощью NumPy.
- Выполнить интерполяцию или аппроксимацию данных.

10. Визуализация данных с Matplotlib: построение графиков (4 ч)

- Построить линейный график зависимости температуры от времени.
- Создать гистограмму распределения значений технологического параметра.
- Добавить подписи осей, заголовков и легенду.

11. Проектная работа: моделирование простого инженерного процесса (6 ч)

- Разработать программу, моделирующую технологический процесс (например, нагрев/охлаждение, движение жидкости).
- Использовать функции, массивы NumPy и визуализацию Matplotlib.
- Подготовить отчёт с описанием алгоритма, кодом и графиками.
- Презентовать результаты работы.

7. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля)

а) основная

1. Марк Лутц, Изучаем Python (4-е издание), Питер, 2018.
2. Эрик Мэтиз, Python для детей. Самоучитель по программированию (адаптировано для начинающих).

б) дополнительная

1. Уэс Маккинни, Python для анализа данных, Питер, 2019.
2. Джон Хант, Введение в научное программирование на Python, 2017.

в) Онлайн-ресурсы

- Официальная документация Python — <https://docs.python.org/3/>
- Документация NumPy — <https://numpy.org/doc/>
- Документация Matplotlib — <https://matplotlib.org/stable/contents.html>

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

Электронные библиотечные системы и электронные библиотеки:
<http://kosgos.ru/nauchnaya-biblioteka.html>

Университетская библиотека ONLINE <https://biblioclub.ru/>

Znanium.com <http://znanium.com/>

Лань <https://e.lanbook.com/>

Электронная библиотека КГУ <http://library.kosgos.ru>

9. Описание материально-технической базы, необходимой для осуществления

образовательного процесса по дисциплине

Для проведения занятий по дисциплине необходимы учебная аудитория? (маркеры для доски), проектор, компьютеры (ноутбуки).