

**МИНОБРНАУКИ РОССИИ**  
**Ярославский государственный университет им. П.Г. Демидова**

Кафедра дифференциальных уравнений

УТВЕРЖДАЮ

Декан математического факультета



Нестеров П.Н.

21 мая 2024 г.

**Рабочая программа дисциплины**  
**Программирование на языке Python**

Направление подготовки (специальности)  
01.03.02 Прикладная математика и информатика

Направленность (профиль)  
«Прикладное программирование и информационные технологии»

Форма обучения очная

Программа рассмотрена  
на заседании кафедры  
от 19 апреля 2024 г., протокол № 8

Программа одобрена НМК  
математического факультета  
протокол № 9 от 3 мая 2024 г.

## 1. Цели освоения дисциплины

Данная дисциплина предполагает изучение языка программирования Python, библиотеки стандартных модулей и принципов разработки программных систем.

Целью освоения дисциплины "Программирование на языке Python" является формирование у студентов навыков, соответствующих видам профессиональной деятельности, необходимых для решения следующих профессиональных задач:

- разработка и применение современных высокопроизводительных вычислительных технологий, применение современных суперкомпьютеров в проводимых исследованиях;
- разработка архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения;
- развитие и использование математических и информационных инструментальных средств, автоматизированных систем в научной и практической деятельности.

## 2. Место дисциплины в структуре образовательной программы

Данная дисциплина относится к обязательной части образовательной программы. Изучение дисциплины продолжает курс информатики старших классов школьной программы и начальных курсов вуза. В ходе программы закрепляются полученные знания изученных ранее курсов «Основы программирования», «Языки программирования». Полученные знания в данном курсе дают очень важные, базисные навыки, в дальнейшем будут использоваться для написания курсовых и дипломных работ и развития программистских навыков обучающихся.

## 3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций в соответствии с ФГОС ВО, ООП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

| Формируемая компетенция<br>(код и формулировка)   | Индикатор достижения компетенции<br>(код и формулировка)  | Перечень планируемых результатов обучения  |
|---|---|--|
| <b>Общепрофессиональные компетенции</b>   |   |  |
| <b>ОПК-5</b><br>Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения | <b>И-ОПК-5.1</b><br>Знает основы программирования, один или несколько языков программирования<br><b>И-ОПК-5.2</b><br>Умеет разрабатывать алгоритмы и компьютерные программы для практического применения<br><b>И-ОПК-5.3</b><br>Имеет практический опыт разработки алгоритмов и компьютерных программ для практического | <b>Знать:</b><br>- основные понятия, виды и характеристики современного программного обеспечения технологии Python;<br>- возможности языка Python;<br>- средства редактирования текста программ;<br>- работа с файлами;<br>- многопоточные приложения;<br>- структуру и принципы функционирования системы программирования;<br>- процесс проектирования и создания компьютерной программы на Python;<br><b>Уметь:</b><br>- проектировать и разрабатывать локальные приложения на языке Python; |

|  |            |  |
|--|------------|--|
|  | применения | <ul style="list-style-type: none"> <li>- использовать библиотеки Python, в том числе устанавливаемые из внешних источников;</li> <li>- использовать среду программирования PyCharm Community для разработки и отладки программ на языке Python</li> </ul> <b>Владеть:</b> <ul style="list-style-type: none"> <li>- основными навыками по созданию программных проектов в IDE;</li> <li>- основными навыками по работе с IDE</li> </ul> |
|--|------------|--|

#### 4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет **2** зачетных единиц, **72** акад. часов.

| №<br>п/п | Темы (разделы)<br>дисциплины,<br>их содержание   | Семестр | Виды учебных занятий,<br>включая самостоятельную<br>работу студентов,<br>и их трудоемкость<br>(в академических часах) |              |              |              |                             |                           | Формы текущего<br>контроля успеваемости<br><br>Форма промежуточной<br>аттестации<br>(по семестрам) |
|----------|--|---------|---|--------------|--------------|--------------|-----------------------------|---------------------------|--|
|          |  |         | Контактная работа   |              |              |              |                             | самостоятельная<br>работа |  |
|          |  |         | лекции  | практические | лабораторные | консультации | аттестационные<br>испытания |                           |  |
| 1        | Вводная лекция.<br>Интерпретатор python.<br>Введение в типы данных<br>python. Динамическая<br>типизация. | 4       | 2   | 4            |              |              |                             | 2                         |  |
| 2        | Числовые типы данных.<br>Списки. Словари. Кортежи.<br>Работа с файлами.                                  | 4       | 2   | 4            |              |              |                             | 2                         |  |
| 3        | Введение в операторы<br>python. Присваивание, ввод-<br>вывод, проверки, циклы,<br>итерации.              | 4       | 2   | 4            |              | 1            |                             | 2                         |  |
| 4        | Функции. Аргументы,<br>области видимости.<br>Включения, генераторы.                                      | 4       | 2   | 4            |              | 1            |                             | 2                         |  |
| 5        | Модули, написание<br>модулей. Пакеты модулей.  | 4       | 2   | 4            |              | 1            |                             | 2                         |  |
| 6        | Объектно-ориентированное<br>программирование в python.<br>Написание классов.<br>Перегрузка операций.     | 4       | 2   | 4            |              | 1            |                             | 2                         |  |
| 7        | Расширенные возможности<br>классов. Исключения.<br>Детали обработки<br>исключений                        | 4       | 2   | 4            |              | 1            |                             | 2                         |  |
| 8        | Управляемые атрибуты.  | 4       | 2   | 4            |              | 1            |                             | 2                         |  |

|  |                        |  |    |    |  |   |     |      |       |
|--|------------------------|--|----|----|--|---|-----|------|-------|
|  | Декораторы. Метаклассы |  |    |    |  |   |     |      |       |
|  |                        |  |    |    |  |   | 0,3 | 1,7  | Зачет |
|  | <b>ИТОГО</b>           |  | 16 | 32 |  | 6 | 0,3 | 17,7 |       |

## 5. Образовательные технологии, в том числе технологии электронного обучения и дистанционные образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

В процессе обучения используются следующие образовательные технологии:

**Вводная лекция** – дает первое целостное представление о дисциплине и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки в целом. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках данной дисциплины, а также дается анализ рекомендуемой учебно-методической литературы.

**Академическая лекция с элементами лекции-беседы** – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Элементы лекции-беседы обеспечивают контакт преподавателя с аудиторией, что позволяет привлекать внимание студентов к наиболее важным темам дисциплины, активно вовлекать их в учебный процесс, контролировать темп изложения учебного материала в зависимости от уровня его восприятия.

**Практическое занятие** – занятие, посвященное освоению конкретных умений и навыков по закреплению полученных на лекции знаний.

**Консультации** – вид учебных занятий, являющийся одной из форм контроля самостоятельной работы студентов. На консультациях по просьбе студентов рассматриваются наиболее сложные моменты при освоении материала дисциплины, преподаватель отвечает на вопросы студентов, которые возникают у них в процессе самостоятельной работы.

## 6. Перечень лицензионного и (или) свободно распространяемого программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине

В процессе осуществления образовательного процесса по дисциплине используются: для формирования материалов для текущего контроля успеваемости и проведения промежуточной аттестации, для формирования методических материалов по дисциплине:

- программы Microsoft Office;
- издательская система LaTeX;
- Adobe Acrobat Reader.

## 7. Перечень современных профессиональных баз данных и информационных справочных систем, используемых при осуществлении образовательного процесса по дисциплине (при необходимости)

В процессе осуществления образовательного процесса по дисциплине используются:

- Автоматизированная библиотечно-информационная система «БУКИ-NEXT»  
[http://www.lib.uniyar.ac.ru/opac/bk\\_cat\\_find.php](http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php)

- Электронно-библиотечная система «Юрайт» <https://urait.ru>
- Электронно-библиотечная система «Лань» <http://e.lanbook.com/>
- Электронно-библиотечная система «Консультант Студента»:  
<https://www.studentlibrary.ru/>

## **8. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет» (при необходимости), рекомендуемых для освоения дисциплины**

### **а) основная литература**

1. Янцев В. В. Web-программирование на Python — Санкт-Петербург: Лань, 2023. <https://reader.lanbook.com/book/310289>
2. Борзунов С. В. Языки программирования. Python: решение сложных задач — Санкт-Петербург: Лань, 2023. <https://reader.lanbook.com/book/319394>

### **б) дополнительная литература**

1. С. В. Борзунов, С. Д. Кургалин Алгебра и геометрия с примерами на Python — Санкт-Петербург: Лань, 2022. <https://reader.lanbook.com/book/202154>

### **в) ресурсы сети «Интернет» (при необходимости)**

<https://docs.python.org/> - официальная документация по языку python

## **9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине**

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

- учебные аудитории для проведения практических занятий (семинаров);
- учебные аудитории для проведения групповых и индивидуальных консультаций;
- учебные аудитории для проведения текущего контроля и промежуточной аттестации;
- помещения для самостоятельной работы;
- помещения для хранения и профилактического обслуживания технических средств обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к электронной информационно-образовательной среде ЯрГУ.

### **Авторы:**

Доцент кафедры дифференциальных уравнений

А.Е. Смирнов

Доцент кафедры дифференциальных уравнений

М.В. Смирнова

**Приложение № 1 к рабочей программе дисциплины  
«Программирование на языке python»**

**Фонд оценочных средств  
для проведения текущего контроля успеваемости  
и промежуточной аттестации студентов  
по дисциплине**

**1. Типовые контрольные задания и иные материалы,  
используемые в процессе текущего контроля успеваемости**

Вариант 1.

Реализуйте класс `Polynom`. Должны быть реализованы следующие функции работы с многочленами (набор функций меняется в зависимости от варианта задания)

- Умножить на другой многочлен
- Сложить с другим многочленом
- Разделить на другой многочлен (если эта операция возможна)
- Вычесть другой многочлен
- Умножить или разделить на число

Вариант 2.

Реализуйте класс `Vector`. Определите для него следующие операции:

- Сложить с другим вектором
- Умножить на число
- Вычесть другой вектор

Вариант 3.

Реализуйте класс `Matrix`. Должен быть реализованы следующие функции работы с матрицами (набор функций меняется в зависимости от варианта задания):

- Умножить на другую матрицу
- Сложить с другой матрицей
- Вычесть другую матрицу
- Умножить/разделить на число

**2. Список вопросов и (или) заданий для проведения промежуточной аттестации**

1. Типы данных в языке python. Изменяемые и неизменяемые типы данных
2. Условные операторы и циклы в python
3. Функции, аргументы функций
4. Модули в python
5. Возможности разработки в объектно-ориентированной парадигме программирования
6. Итераторы
7. Генераторы
8. Исключения
9. Декораторы
10. Метаклассы

## **Приложение № 2 к рабочей программе дисциплины «Программирование на языке python»**

### **Методические указания для студентов по освоению дисциплины**

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций», или других заданий аналогичного уровня сложности