

**Аннотация рабочей программы дисциплины (модуля)**  
**Б1.В.ДВ.03.01 Программирование на языке Python**

**Цели дисциплины**

Целями освоения дисциплины Программирование на языке Python является формирование профессиональных компетенций будущих специалистов в области прикладной математики, изучение основных методов построения программных прикладных продуктов на языке Python.

**Задачи дисциплины**

Основными задачами изучения дисциплины являются:

- формирование навыков и умений программирования на языке общего назначения Python, ориентированного на повышение производительности разработчика и читаемости кода;
- овладение знаниями, умениями и навыками по организации разработки приложений для вычислительной аналитики;
- формирование умений проектирования приложений, навыков их практической реализации.

**Формируемые компетенции и индикаторы их достижения по дисциплине**

<b>Код компетенции</b>	<b>Содержание компетенции</b>	<b>Код и наименование индикатора достижения компетенции</b>
ПКС-2	Способен проектировать структуры данных	ПКС-2.1 Знать основные принципы проектирования структур данных. ПКС -2.2 Уметь использовать принципы проектирования структур данных. ПКС-2.3 Иметь навыки проектирования структур данных.
ПКС-5	Способен понимать, совершенствовать и применять современный математический аппарат	ПКС - 5.1 Обладает базовыми знаниями, полученными в области математических и естественных наук и методами решения задач профессиональной деятельности с использованием существующих информационно-коммуникационных технологий. ПКС - 5.2 Умеет решать стандартные задачи в научно-исследовательской и профессиональной деятельности с использованием существующих информационно-коммуникационных технологий. ПКС - 5.3 Имеет практические навыки в области решения стандартных задач в научно-исследовательской и профессиональной деятельности с использованием существующих информационно-коммуникационных технологий

**Содержание разделов дисциплины**

**Тема 1. Язык Python. Типы данных, операции, операторы. Особенности ввода/вывода.**  
Синтаксис языка Python для основных алгоритмических конструкций, литералов,

выражений. Описание встроенных типов данных, особенности общепринятого в Python стиля программирования.

## **Тема 2. Встроенные типы объектов: числа, строки, кортежи, списки, словари, множества**

Разнообразие типов-коллекций в языке Python. Общие подходы и особенности при их использовании при написании программ: индексирование, срезы, функции вычисления длины, максимального и минимального значений, сортировки, смены порядка следования элементов и др. Особенности каждого из типов данных.

## **Тема 3. Файловый ввод-вывод. Чтение строк с помощью итераторов файлов. Работа с двоичными файлами.**

Функции открытия, закрытия, чтения, записи в текстовый файл. Обработка файлов в формате UNICODE. Использование итераторов при работе с файлами. Хранение и обработка информации при помощи двоичных файлов.

## **Тема 4. Библиотека numpy для реализации математических объектов и вычислений.**

Пакет Numpy для осуществления численных расчетов и выполнения матричных вычислений, обзор других пакетов для научных вычислений.

## **Тема 5. Создание приложений с GUI. Обзор графических библиотек: Tkinter, PyQt.**

Приложения с графическим интерфейсом пользователя. Использование возможностей графической библиотеки виджетов (Tk) для построения интерфейса вместо визуальных средств ("построители интерфейса").

## **Тема 6. Классы в Python. Определение данных, методов, операций. Наследование. Множественное наследование. Композиция при разработке классов.**

Особенности реализации трех принципов ООП в Python: инкапсуляция, наследование, полиморфизм. Отношения между классами: наследование, ассоциация, агрегация. Статические методы, мультиметоды, устойчивые объекты.