

**ГОУ ВПО Российско-Армянский (Славянский)
университет**

Утверждено
Директор Института математики и информатики
Арамян Р.Г.



«21» мая 2025, протокол №9.1

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ

Наименование дисциплины: Основы Python

Автор: Хачатуров Николас Арменович

Направление подготовки: 01.03.02 Прикладная математика и информатика
Наименование образовательной программы: 01.03.02 Прикладная математика и информатика

1. АННОТАЦИЯ

1.1. Краткое описание содержания данной дисциплины;

Курс "Основы Python" предназначен для студентов с хорошими навыками программирования и стремятся погрузиться в область машинного обучения с помощью Python. В рамках этого курса студенты углубят свои знания, изучив ключевые библиотеки Python, необходимые для машинного обучения, и приобретут навыки реализации разнообразных алгоритмов и моделей, используя библиотеку scikit-learn.

1.2. Трудоемкость в академических кредитах и часах, формы итогового контроля (экзамен/зачет); 4 кредита, ЗаО, 144 часа

1.3. Взаимосвязь дисциплины с другими дисциплинами учебного плана специальности (направления)

Данная дисциплина логически и содержательно методически связана с такими дисциплинами как «Основы программирования», «Теория алгоритмов и математическая логика», «Объектно-ориентированное программирование», «Основы программирования». Является логически связанной с математическими дисциплинами, рассматривает объекты таких дисциплин как «Дискретная математика» с точки зрения программирования.

1.4. Результаты освоения программы дисциплины:

Код компетенции	Наименование компетенции	Код индикатора достижения компетенций	Наименование индикатора достижений компетенций
ПК-1	способностью собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям	1	Знать принципы определения актуальности и практической значимости НИР на основе обобщения, анализа
		2	Уметь работать с научными источниками, проводить анализ и критически оценивать результаты научных исследований
		3	Владеть опытом выделять сильные и слабые стороны,

			определять значимость научных источников
ПК-5	способностью осуществлять целенаправленный поиск информации о новейших научных и технологических достижениях в информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет") и в других источниках	1	знать основные референтные базы данных научных публикаций, поисковые системы научной литературы
		2	уметь осуществлять поиск научной литературы с использованием существующих поисковых систем и референтных баз данных
		3	владеть навыками поиска научной литературы
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	1	Знает подходы в постановке задач для достижения поставленной цели, обладает знаниями в выборе оптимальных способов их решения
		2	Умеет, исходя из действующих правовых норм, имеющихся ресурсов и ограничений, выбирать оптимальные способы решения задач в профессиональной области для достижения поставленной цели
		3	Владеет навыками определения круга профессиональных задач в

			рамках поставленной цели; выбором оптимальных способов их решения с учетом действующих правовых норм и имеющихся ресурсов
--	--	--	---

2. УЧЕБНАЯ ПРОГРАММА

2.1. Цели и задачи дисциплины

2.2. Трудоемкость дисциплины и виды учебной работы (в академических часах и зачетных единицах) *(удалить строки, которые не будут применены в рамках дисциплины)*

Виды учебной работы	Всего, в акад. часах	Распределение по семестрам					
		5 сем	— сем	— сем	— сем.	— сем	— сем.
1	2	3	4	5	6	7	8
1. Общая трудоемкость изучения дисциплины по семестрам, в т. ч.:							
1.1. Аудиторные занятия, в т. ч.:							
1.1.1. Лекции	32	32					
1.1.2. Практические занятия, в т. ч.	32	32					
1.2. Самостоятельная работа, в т. ч.:	80	80					
1.2.1. Подготовка к экзаменам							
Итоговый контроль (диф. зачет)							

2.3. Содержание дисциплины

2.3.1. Тематический план и трудоемкость аудиторных занятий (модули, разделы дисциплины и виды занятий) по рабочему учебному плану

Разделы и темы дисциплины	Всего (ак. часов)	Лекции (ак. часов)	Практ. Занятия (ак. часов)	Семинары (ак. часов)	Лабор. (ак. часов)
1	2=3+4+5+6 +7	3	4	5	6
Тема 1. Введение в Python	4	2	2		
Тема 2. Переменные и типы данных	4	2	2		
Тема 3. Условия и циклы	4	2	2		

Тема 4. Коллекции: списки, кортежи, словари, множества	4	2	2		
Тема 5. Функции	4	2	2		
Тема 6: Классы и объектно-ориентированное программирование	4	2	2		
Тема 7. Обработка исключений	4	2	2		
Тема 8. Работа с файлами и файловой системой	4	2	2		
Тема 9. Тестирование кода	4	2	2		
Тема 10. Установка библиотек и pip	4	2	2		
Тема 11. Виртуальное окружение	4	2	2		
Тема 12. Визуализация данных: Matplotlib и Seaborn	4	2	2		
Тема 13. Работа с математическими объектами: NumPy	4	2	2		
Тема 14. Работа с таблицами: Pandas	6	3	3		
Тема 15. Работа с изображениями и видео: OpenCV	6	3	3		
ИТОГО	64	32	32		

2.3.2. Краткое содержание разделов дисциплины в виде тематического плана

Тема 1. Введение в Python

Рассматриваются история и особенности языка Python, области его применения, установка и запуск, работа в интерактивной оболочке и написание первой программы.

Тема 2. Переменные и типы данных

Изучаются переменные, правила их именования и основные типы данных: числа, строки, логические значения. Рассматриваются операции преобразования типов и базовый ввод-вывод.

Тема 3. Условия и циклы

Рассматриваются условные конструкции if, elif, else, циклы for и while, операторы управления потоком выполнения: break, continue, pass.

Тема 4. Коллекции: списки, кортежи, словари, множества

Изучаются структуры данных: списки, кортежи, множества и словари. Рассматриваются основные операции и методы работы с каждым типом коллекций.

Тема 5. Функции

Рассматривается определение и вызов функций, работа с аргументами и возвращаемыми значениями, параметры по умолчанию, а также анонимные функции (lambda) и области видимости.

Тема 6. Классы и объектно-ориентированное программирование

Изучаются основы ООП: классы, объекты, инкапсуляция, наследование и полиморфизм. Рассматривается создание собственных классов и взаимодействие между объектами.

Тема 7. Обработка исключений

Рассматриваются конструкции try, except, finally, else для обработки ошибок. Изучаются типовые исключения и их перехват, создание собственных исключений.

Тема 8. Работа с файлами и файловой системой

Изучается работа с файлами: открытие, чтение, запись, использование with. Рассматриваются базовые операции с директориями и путями через встроенные модули.

Тема 9. Тестирование кода

Изучаются принципы написания и автоматизации тестов. Рассматриваются типы тестов, работа с модулем unittest, проверка результатов и обработка ошибок.

Тема 10. Установка библиотек и pip

Рассматриваются основы работы с менеджером пакетов pip, установка и обновление внешних библиотек, создание и использование файла зависимостей requirements.txt.

Тема 11. Виртуальное окружение

Изучается создание и использование виртуальных окружений для изоляции проектов. Рассматриваются команды создания, активации, установки и сохранения зависимостей.

Тема 12. Визуализация данных: Matplotlib и Seaborn

Рассматриваются основы построения графиков в matplotlib и расширенные средства визуализации в seaborn: распределения, зависимости, тепловые карты и др.

Тема 13. Работа с математическими объектами: NumPy

Изучаются массивы NumPy, операции с ними, трансформации, индексирование и математические вычисления. Рассматриваются универсальные функции и генерация данных.

Тема 14. Работа с таблицами: Pandas

Рассматривается работа с табличными структурами DataFrame и Series: чтение данных, фильтрация, агрегация, обработка пропущенных значений и сортировка.

Тема 15. Работа с изображениями и видео: OpenCV

Изучается загрузка, отображение и сохранение изображений. Рассматриваются операции над пикселями, преобразование цветовых пространств, работа с видео и веб-камерой.

2.3.3. Краткое содержание семинарских/практических занятий/лабораторного практикума

Семинарские занятия проводятся в практико-ориентированной форме. Студентам предоставляется набор задач, соответствующих тематике занятия. Часть задач решается

непосредственно в аудитории под руководством преподавателя, с пошаговым разбором решений и обсуждением используемых подходов. Оставшиеся задачи выдаются студентам в качестве домашнего задания для самостоятельного выполнения.

2.3.4. Материально-техническое обеспечение дисциплины

Для проведения занятий используется компьютерный класс, оснащённый современными персональными компьютерами с установленной операционной системой Windows или Linux. На рабочих местах студентов и преподавателя должны быть установлены интерпретатор Python, среда разработки VS Code.

2.4. Модульная структура дисциплины с распределением весов по формам контролей

Формы контролей	Вес формы (форм) текущего контроля в результирующей оценке текущего контроля (по модулям)		Вес формы промежуточного контроля в итоговой оценке промежуточного контроля		Вес итоговой оценки промежуточного контроля в результирующей оценке промежуточных контролей		Вес итоговой оценки промежуточного контроля в результирующей оценке промежуточных контролей (семестровой оценке)		Весы результирующей оценки промежуточных контролей и оценки итогового контроля в результирующей оценке итогового контроля
	M1 ¹	M2	M1	M2	M1	M2			
Вид учебной работы/контроля									
Контрольная работа <i>(при наличии)</i>									
Устный опрос <i>(при наличии)</i>									
Тест <i>(при наличии)</i>									
Лабораторные работы <i>(при наличии)</i>									
Письменные домашние задания <i>(при наличии)</i>									
Реферат <i>(при наличии)</i>									
Эссе <i>(при наличии)</i>									
Проект <i>(при наличии)</i>									
Другие формы <i>(при наличии)</i>									

¹ Учебный Модуль

Веса результирующих оценок текущих контролей в итоговых оценках промежуточных контролей								
Веса оценок промежуточных контролей в итоговых оценках промежуточных контролей								
Вес итоговой оценки 1-го промежуточного контроля в результирующей оценке промежуточных контролей							0.5	
Вес итоговой оценки 2-го промежуточного контроля в результирующей оценке промежуточных контролей							0.5	
Вес результирующей оценки промежуточных контролей в результирующей оценке итогового контроля								0.4
Вес итогового контроля (Экзамен/зачет) в результирующей оценке итогового контроля								0.6
	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$	$\Sigma = 1$

3. Теоретический блок

3.1. Материалы по теоретической части курса

3.1.1. CS50's Introduction to Programming with Python (edX);

3.1.2. Google's Python Class;

3.1.3. Книга «Изучаем Python» — Марк Лутц;

4. Фонды оценочных средств

Во время проверки знаний студенту будет предложена **практическая задача**, которую необходимо решить, используя Python и соответствующие библиотеки. Задача будет направлена на проверку как технических навыков, так и понимания основ работы с данными.

Как проходит оценивание:

1. Выдача задачи

Студент получает задачу, которую нужно решить за ограниченное время. Решение должно быть представлено в виде кода с объяснениями при необходимости.

2. Пример задачи

"Дан CSV-файл с данными о продажах: `sales.csv`. Рассчитайте общее количество продаж по каждому товару и найдите товар с наибольшим стандартным отклонением по количеству продаж по дням."

3. Проверка корректности решения

После того как студент представит решение, оно будет проверено на:

- правильность результата;
- эффективность и читаемость кода;
- корректное использование выбранных библиотек.

4. Устные вопросы по решению

После сдачи задачи студенту будет задано **несколько уточняющих вопросов**, например:

- Почему вы выбрали именно эту функцию/метод?
- Что делает конкретная строка кода?
- Как изменится результат, если входные данные будут содержать пропуски?

Это позволяет не только оценить знание синтаксиса, но и глубину понимания работы с данными.

5. Методический блок

Преподавание данной учебной дисциплины строится на сочетании лекций, практических занятий и различных форм самостоятельной работы студентов.

На лекциях рассматриваются основные концепции и ключевые инструменты языка Python, с акцентом на наиболее сложных аспектах, таких как работа с данными, объектно-ориентированное программирование и модульная структура программ. Особое внимание уделяется практическим примерам и разбору реальных кейсов, что позволяет студентам глубже понять принципы написания чистого, читаемого и эффективного кода.

На практических занятиях акцент делается на применение полученных знаний для решения конкретных задач. Студенты осваивают навыки:

- написания и отладки программ;
- работы с разными библиотеками;
- реализации алгоритмов и структур данных;
- чтения и анализа чужого кода.

При проведении семинарских и практических занятий особое внимание уделяется:

- развитию логического и алгоритмического мышления;
- формированию навыков написания кода, соответствующего стандартам индустрии;
- обучению эффективному поиску и использованию справочной информации и документации.

Рабочей программой дисциплины предусмотрена самостоятельная работа студентов с целью углубления знаний и формирования практических навыков. Она включает:

- изучение и повторение лекционного материала;
- выполнение домашних заданий и мини-проектов;
- чтение дополнительной литературы и официальной документации;
- подготовку к контрольным точкам, зачётам и экзаменам;
- участие в код-ревью и обсуждении решений в учебных группах.

Такой подход обеспечивает системное освоение языка Python и позволяет студентам уверенно использовать его как в академических, так и в практико-ориентированных задачах.