

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

Утверждено
Директор Института Математики и Информатики
Арамян Р.Г.



«21» марта 2025, протокол № 9.1

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ДИСЦИПЛИНЫ

Наименование дисциплины: Framework(R / Python)

Автор : Аванян Анна Беньяминовна

Направление подготовки: 01.04.02 Прикладная математика и информатика

Наименование образовательной программы: Машинное обучение и наука о данных

1. АННОТАЦИЯ

1.1. Краткое описание содержания данной дисциплины

В курсе «Framework(R / Python)» студенты должны знать:

- синтаксис языка программирования Python;
- основные принципы объектно-ориентированного программирования.
- основные классы из библиотеки классов языка программирования Python для создания объектно-ориентированных приложений. уметь:
- разрабатывать программы на языке программирования Python, создавая собственные классы, а также использовать классы и модули из библиотек этого языка;
- создавать удобный интерфейс для использования созданных программных средств с помощью библиотеки TkInter или др. владеть:
- навыками разработки консольных приложений в стиле объектно-ориентированного программирования на языке программирования Python;
- навыками разработки приложений с внешними источниками данных (текстовыми файлами, xml-файлами, базами данных);
- использовать набор библиотек языка Python для научных вычислений и научной визуализации демонстрировать способность и готовность:
- применять полученные знания и навыки в процессе дальнейшего обучения, при написании курсовых и выпускных работ, а также в своей дальнейшей профессиональной деятельности.

1.2. Трудоемкость в академических кредитах и часах, формы итогового контроля

Трудоемкость курса составляет 5 академических кредита, что эквивалентно 180 часам (1 академический кредит равен 36 часам). Итоговый контроль будет проведен в форме **зачета**.

1.3. Взаимосвязь дисциплины с другими дисциплинами учебного плана специальности

При изучении дисциплины «Introduction to ML» используются понятия и методы численных методов оптимизации, дискретного анализа и т.д.

1.4. Результаты освоения программы дисциплины:

Код компетенции	Наименование компетенции	Код индикатора достижения компетенций	Наименование индикатора достижений компетенций
УК-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, выработать стратегию действий	УК-1.1	Знать теории и методы критического анализа и системного подхода
		УК-1.2	Уметь применять методы поиска, анализа и синтеза информации для решения конкретных задач
		УК-1.3	Владеть способностью выработки стратегических решений на основе комплексного анализа ситуации и прогнозирования исходов
УК-3	Способен организовывать и руководить работой команды, выработывая командную стратегию для достижения поставленной цели	УК-3.1	Знать основы социального взаимодействия и теории командной работы
		УК-3.2	Уметь эффективно коммуницировать и сотрудничать в команде
		УК-3.3	Владеть навыками стратегического планирования и лидерства, способными максимизировать потенциал команды для достижения общих целей
УК-4	Способен применять современные коммуникативные технологии, в том числе на иностранном(ых) языке(ах), для академического и	УК-4.1	Знать ключевые коммуникативные технологии и инструменты, используемые для международного и

	профессионального взаимодействия		межкультурного общения
		УК-4.2	Уметь эффективно использовать технологии для общения и сотрудничества в многоязычной и мультикультурной среде
		УК-4.3	Владеть мастерством межкультурной коммуникации, умением адаптировать сообщения для различных культурных и профессиональных контекстов
ПК-1	Способностью проводить научные исследования и получать новые научные и прикладные результаты самостоятельно и в составе научного коллектива	ПК-1.1	Знать методы и подходы научных исследований в прикладной математике и информатике
		ПК-1.2	Уметь проводить научные исследования самостоятельно и в составе коллектива
		ПК-1.3	Владеть умением организовывать и руководить научными проектами, обеспечивая получение новых результатов

2. УЧЕБНАЯ ПРОГРАММА

2.1. Цели и задачи дисциплины

Цели дисциплины:

- 1. Овладение специализированным программным инструментом:** Обучение студентов эффективному использованию выбранного языка программирования (R или Python) в контексте анализа данных и машинного обучения.
- 2. Развитие практических навыков:** Подготовка специалистов, способных применять программные навыки в реальных исследовательских и промышленных проектах.
- 3. Интеграция теоретических и практических знаний:** Помочь студентам связать теоретические основы статистики и анализа данных с практическими задачами через программирование.

Задачи дисциплины:

- 1. Изучение базовых и продвинутых функций:** Подробное изучение основных и специализированных пакетов и библиотек для R или Python, в зависимости от выбора студента.
- 2. Работа с данными:** Научить студентов методам предварительной обработки, анализа и визуализации данных.
- 3. Программирование и скриптинг:** Обучение написанию эффективных скриптов и программ, которые могут автоматизировать сбор, обработку и анализ данных.
- 4. Реализация статистических методов:** Применение статистических методов и машинного обучения для анализа данных с использованием R или Python.

2.2. Трудоемкость дисциплины и виды учебной работы

Виды учебной работы	Всего, в акад. часах	Распределение по семестрам					
		<u>1</u> сем	<u> </u> сем	<u> </u> сем	<u> </u> сем.	<u> </u> сем	<u> </u> сем.
1	2	3	4	5	6	7	8
1.Общая трудоемкость изучения дисциплины по семестрам, в т. ч.:	32	32					
1.1.Аудиторные занятия, в т. ч.:	32	32					
1.1.1.Лекции							
1.1.2.Практические занятия, в т. ч.	32	32					
Итоговый контроль		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 часов	Лекции, часов	Практ. занятия, часов	Семинары, часов	Лабор. часов	Другие виды занятий, часов
1	2	3	4	5	6	7

I курс						
МОДУЛИ.	32	32				
Вводное занятие. История и преимущества Python.	2	2				
Тема 1. Переменные. Переменные в математике, программировании, Python.	2	2				
Тема 2. Типы данных. Числа. Общая информация, типы данных в Python	2	2				
Тема 3. Условная инструкция. Общая информация, условия в Python	2	2				
Тема 4. Циклы. Общая информация, Циклы for и while, операторы break и continue.	4	4				
Тема 5. Типы данных. Строки. Общая информация, строки в Python	4	4				
Тема 6. Типы данных. Последовательности. Список, кортеж.	2	2				
Тема 7. Типы данных. Последовательности. Множество, операторы членства.	2	2				
Тема 8. Функции. Общая информация, функции в Python.	2	2				
Тема 9. Вводное занятие. Ингредиенты Python. Конструирование инструкций. Области Видимости.	2	2				
Тема 10. Словари. Работа с файлами.	2	2				
Тема 11. Исключения. Логирование.	2	2				
Тема 12. Data science/ML пользовательский интерфейс..	2	2				
Тема 13. Введение в ООП Python. Наследование. Абстракция. Инкапсуляция. Полиморфизм.	2	2				
ИТОГО	32	32				

2.3.2. Краткое содержание разделов дисциплины в виде тематического плана

Тема 1. Вводное занятие. История и преимущества Python.

- Введение в Python, его история развития и основные преимущества. Обзор сфер применения и популярности языка.

Тема 2. Переменные. Переменные в математике, программировании, Python.

- Основы работы с переменными в Python, сравнение с математическими переменными, особенности типизации и использования в программировании.

Тема 3. Типы данных. Числа. Общая информация, типы данных в Python.

- Изучение числовых типов данных в Python, включая целые числа, числа с плавающей точкой; основы числовых операций.

Тема 4. Условная инструкция. Общая информация, условия в Python.

- Обзор условных операторов в Python, включая if, elif, и else; практические примеры использования условных инструкций.

Тема 5. Циклы. Общая информация, Циклы for и while, операторы break и continue.

- Основы работы с циклами в Python, включая for и while, а также использование операторов break и continue для контроля потока выполнения.

Тема 6. Типы данных. Строки. Общая информация, строки в Python.

- Работа со строковыми типами данных, методы и операции над строками, их применение в реальных задачах.

Тема 7. Типы данных. Последовательности. Список, кортеж.

- Обзор последовательностей в Python, в том числе списков и кортежей, их свойства и методы обработки.

Тема 8. Типы данных. Последовательности. Множество, операторы членства.

- Изучение множеств и операторов членства, особенности их использования для управления уникальными элементами.

Тема 9. Функции. Общая информация, функции в Python.

- Подробный обзор создания и использования функций в Python, параметры, возвращаемые значения и области видимости.

Тема 10. Вводное занятие. Ингредиенты Python. Конструирование инструкций. Области видимости.

- Введение в структуры данных Python, их интеграция в программы и управление областями видимости.

Тема 11. Словари. Работа с файлами.

- Работа со словарями и файловой системой Python, основы чтения и записи файлов, использование словарей для организации данных.

Тема 12. Исключения. Логирование.

- Управление исключениями в Python, основы логирования для отслеживания и диагностики в программе.

Тема 13. Data science/ML пользовательский интерфейс.

- Основы использования Python в областях Data Science и машинного обучения, работа с пользовательскими интерфейсами.

Тема 14. Введение в ООП Python. Наследование. Абстракция. Инкапсуляция. Полиморфизм.

- Обзор объектно-ориентированного программирования в Python, включая наследование, абстракцию, инкапсуляцию и полиморфизм.

2.3.3. Краткое содержание семинарских/практических занятий/лабораторного практикума

• Лабораторные работы:

- **Применение:** Студенты выполняют задания в лабораторных условиях, где они практикуются в написании кода, обработке данных, и создании функций в выбранном языке программирования (R или Python).
- **Цель:** Научиться реализовывать алгоритмы, обрабатывать и анализировать данные с использованием различных библиотек и пакетов.

• Проектные работы:

- **Применение:** Разработка и реализация конкретных проектов, где студенты используют свои навыки для решения реальных задач, таких как визуализация данных, статистический анализ или машинное обучение.
- **Цель:** Применить теоретические знания на практике, развить навыки решения проблем и умение работать над комплексными проектами.

2.3.4. Материально-техническое обеспечение дисциплины

Компьютеры:

- Наличие компьютеров, оборудованных достаточной оперативной памятью и процессорной мощностью для эффективной работы с аналитическим и программным обеспечением.

- Установленные операционные системы и необходимое программное обеспечение, включая последние версии R и Python, а также дополнительные библиотеки и инструменты для данных языков.

Проектор:

- Проектор для визуализации учебных материалов, презентаций и кода во время практических занятий.
- Обеспечение возможности демонстрации программного кода и аналитических результатов для больших групп студентов.

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)		Веса 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>									

¹ Учебный Модуль

<i>Другие формы (при наличии)</i>								
Весы результирующих оценок текущих контролей в итоговых оценках промежуточных контролей						0,3		
Весы оценок промежуточных контролей в итоговых оценках промежуточных контролей						0,7		
Вес итоговой оценки 1-го промежуточного контроля в результирующей оценке промежуточных контролей								
Вес итоговой оценки 2-го промежуточного контроля в результирующей оценке промежуточных контролей							1	
Вес результирующей оценки промежуточных контролей в результирующей оценке итогового контроля								1
Вес итогового контроля (Экзамен/зачет) в результирующей оценке итогового контроля								
	$\Sigma = 1$							

3. Теоретический блок *(указываются материалы, необходимые для освоения учебной программы дисциплины)*

3.1. Материалы по теоретической части курса

3.1.1. Учебник(и);

1. Сузи, Р.А. Язык программирования Python, 2007.
2. Sweigart, A. Разработка компьютерных игр на языке Python, 2016.

4. Фонды оценочных средств *(указываются материалы, необходимые для проверки уровня знаний в соответствии с содержанием учебной программы дисциплины).*

4.1. Планы практических и семинарских занятий

Практические задания: Разработка и выполнение заданий, которые требуют применения программных навыков для решения реальных аналитических задач. Задания могут включать анализ данных, визуализацию результатов, разработку алгоритмов.

Семинары: Обсуждение и анализ теоретического материала и практических примеров. В ходе семинаров студенты представляют результаты своих работ, которые затем обсуждаются с преподавателем и другими студентами.

4.2. Планы лабораторных работ и практикумов

Лабораторные работы: Осуществление практических занятий в компьютерном классе, где студенты, работая индивидуально или в группах, выполняют задания по программированию, анализу данных и использованию специализированных библиотек R или Python.

Практикумы: Глубокое изучение специализированных тем и технологий, применение сложных методов анализа данных, работа с большими массивами данных и реализация проектов машинного обучения.

Материалы для оценки:

Контрольные работы: Проведение письменных работ, направленных на проверку теоретических знаний студентов по ключевым аспектам курса.

Тесты: Организация тестирования для оценки уровня освоения программных навыков и понимания основных принципов работы с R или Python.

Проекты: Оценка индивидуальных или групповых проектов, которые показывают способность студентов применять навыки в реальных условиях.

5. Методический блок

5.1. Методика преподавания

Методика преподавания курса направлена на интеграцию теоретических знаний и практических умений. Она организована таким образом, чтобы стимулировать активное и самостоятельное изучение, а также развивать навыки решения реальных задач с помощью программирования.

5.1.1. Методические рекомендации для студентов по подготовке к семинарским, практическим или лабораторным занятиям, по организации самостоятельной работы студентов при изучении конкретной дисциплины.

- Подготовка к занятиям:
- Предварительное изучение материалов
- Организация самостоятельной работы
- Регулярное выполнение
- Подготовка к практическим и лабораторным работам:
- Практика программирования.