

## АННОТАЦИЯ ДИСЦИПЛИНЫ

### «Программирование (Python)»

Дисциплина «Программирование (Python)» является частью программы магистратуры «Автоматизированные системы обработки информации и управления» по направлению «09.04.01 Информатика и вычислительная техника».

#### Цели и задачи дисциплины

Формирование комплекса знаний, умений и навыков в области применения языка программирования Python для разработки прикладного программного обеспечения, выполнения научных вычислений, интеллектуального анализа данных, разработки интеллектуальных систем на основе методов машинного обучения..

#### Изучаемые объекты дисциплины

Язык программирования Python; стандартная библиотека языка Python; дополнительные модули и библиотеки научных вычислений; модули и библиотеки анализа данных; модули и библиотеки визуализации данных; модули и библиотеки машинного обучения; модули и библиотеки разработки веб-приложений..

#### Объем и виды учебной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах			
		Номер семестра			
		1			
1. Проведение учебных занятий (включая проведение текущего контроля успеваемости) в форме:	54	54			
1.1. Контактная аудиторная работа, из них:					
- лекции (Л)				18	18
- лабораторные работы (ЛР)				18	18
- практические занятия, семинары и (или) другие виды занятий семинарского типа (ПЗ)				16	16
- контроль самостоятельной работы (КСР)				2	2
- контрольная работа					
1.2. Самостоятельная работа студентов (СРС)	54	54			
2. Промежуточная аттестация					
Экзамен	36	36			
Дифференцированный зачет					
Зачет					
Курсовой проект (КП)					
Курсовая работа (КР)					
Общая трудоемкость дисциплины	144	144			

#### Краткое содержание дисциплины

Наименование разделов дисциплины с кратким содержанием	Объем аудиторных занятий по видам в часах			Объем внеаудиторных занятий по видам в часах
	Л	ЛР	ПЗ	СРС
1-й семестр				
Анализ данных и машинное обучение на Python	4	6	4	8
Современные задачи интеллектуального анализа данных и машинного обучения. Библиотеки научных вычислений, анализа данных и визуализации: NumPy, SciPy, matplotlib, pandas, scikit-learn, seaborn. Фреймворки машинного обучения PyTorch, Keras, TensorFlow. Библиотеки компьютерного зрения. Границы применимости, достоинства и недостатки.				
Функциональное программирование в Python	2	2	2	8
Принципы декларативного и функционального программирования. Интерпретаторы и генераторы. Списковые включения. Словарные включения. Lambda-выражения (анонимные функции). Функции Map, Reduce и Filter. Функции высшего порядка. Частичное исполнение функции. Модуль functools.				
Императивное программирование в Python	2	0	2	4
Структура Python-модуля. Скрипты. Поведение модуля при импорте в другие модули. Области видимости переменных. PEP8 — руководство по оформлению кода. Достоинства и недостатки PEP8.				
Тестирование и развертывание кода на Python	2	2	0	8
Виртуальная среда разработки. Структура проекта на Python. Модули. Иерархия классов. Модульное тестирование. Модуль unittest. Интеграционное тестирование. Selenium. Контейнеризация Python-проектов. Микросервисы.				
Объектно-ориентированное программирование в Python	2	2	0	8
Реализация классов стандартной библиотеки. Принцип «объект = функция, функция = объект». Имитация функции. Особенности именования методов классов. «Магические» методы. Пользовательские классы.				

Наименование разделов дисциплины с кратким содержанием	Объем аудиторных занятий по видам в часах			Объем внеаудиторных занятий по видам в часах
	Л	ЛР	ПЗ	СРС
Асинхронное и параллельное программирование в Python	2	2	4	8
Понятие асинхронного выполнения кода. Понятие параллельного исполнения программы. Модули стандартной библиотеки threading и multiprocessing. Понятие Global Interpreter Lock (GIL). Программы, управляемые событиями.				
Основы языка Python	2	2	2	4
Понятие интерпретируемого языка программирования. Установка и настройка Python. Режимы работы Python. iPython. Проект Jupyter. Формат Jupyter Notebooks. Conda. Типы данных. Коллекции. Использование списков, кортежей, множеств, словарей.				
Веб-разработка на Python	2	2	2	6
Базовые концепции разработки веб-приложений. Понятия фронтенда и бэкенда. Веб-фреймворки на Python: Django, Flask, FastAPI. Интеграция веб-приложения с другими модулями и библиотеками Python. Интеграция Python-бэкенда с JavaScript- и TypeScript-фреймворками фронтенда.				
ИТОГО по 1-му семестру	18	18	16	54
ИТОГО по дисциплине	18	18	16	54