

«УТВЕРЖДАЮ»
 Директор Института цифровых
 технологий и математики
 _____ С.А. Рощупкин



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.01.10 Предметно-ориентированное программирование

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Компьютерное моделирование и анализ данных

Квалификация (степень): бакалавр

Форма обучения: очная

Институт: цифровых технологий и математики

Кафедра: математического моделирования, компьютерных технологий и информационной безопасности

	очная форма	очно-заочная форма	заочная форма
Курс	3,4		
Семестр	6,7		

Лекции	28		
Лабораторные занятия	56		
Практические (семинарские) занятия	38		
в т. ч. практическая подготовка	8		
Форма(ы) промежуточной аттестации	Зачет – 6 семестр Экзамен - 0,5 КП - 0,5		
Контроль	9		
Иные формы работы	-		
Самостоятельная работа	155,2		

Всего часов: 288

Трудоемкость: 8 зачетных единиц.

Разработчик(и) рабочей программы:

Ассистент

Маторин Д.Д.

I. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЙ РАЗДЕЛ

Цель изучения дисциплины:

Формирование и использование систематизированных теоретических и практических знаний при решении профессиональных задач в области анализа данных и компьютерного моделирования с применением языка программирования Python.

Задачи изучения дисциплины:

- Изучить методы работы с объектами языка Python, используемыми для аналитики данных и компьютерного моделирования сложных систем.
- Освоить проектирование и обработку Big Data с использованием экосистемы Python (Pandas, NumPy, Dask).
- Изучить способы и средства проектирования программных интерфейсов (API, функции, классы, модули).
- Сформировать представление о комплексном использовании системы компоновки и обработки данных в научных и прикладных задачах.

Место дисциплины в структуре ОПОП: реализуется в рамках части, формируемой участниками образовательных отношений, блока Б1. Дисциплины (модули).

Планируемые результаты обучения по дисциплине:

Код компетенции	Индикаторы достижения компетенции	Планируемые результаты обучения по дисциплине
ПКС-1	Знать: <ul style="list-style-type: none">– возможности существующей программно-технической архитектуры;– методологию разработки программного обеспечения и технологию программирования;– методы и средства проектирования программного обеспечения;– типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения.	Знает: <ul style="list-style-type: none">– методы формализации задач;– приемы работы с объектами Python (DataFrame, массивы, классы);– экосистему библиотек для анализа данных (Pandas, NumPy, SciPy, Scikit-learn);– основы ООП и функционального программирования в Python.
	Уметь: <ul style="list-style-type: none">– проводить оценку и обоснование рекомендуемых решений;– вырабатывать варианты реализации программного обеспечения;	Умеет: <ul style="list-style-type: none">– применять методы проектирования ПО и структур данных в Python;– разрабатывать интерфейсы для анализа данных;– проектировать и реализовывать пай-

	– применять методы и средства проектирования программного обеспечения, структур данных, программных интерфейсов.	плайны обработки данных; – использовать векторизацию и оптимизацию кода
	Владеть: – анализом возможностей реализации требований к программному обеспечению; – навыками распределения заданий между программистами в соответствии с техническими спецификациями; – методами проектирования структур данных; – методами проектирования программных интерфейсов; – навыками осуществления обучения и наставничества.	Владеет: – инструментами разработки и модификации архитектуры ПО на Python; – навыками проектирования структур данных (DataFrame, массивы, словари); – методами работы с Big Data (Dask, chunking); – средствами проектирования API и модульных систем; – навыками разработки технической документации; способностью оценивать сроки выполнения задач

II. СОДЕРЖАНИЕ И ОБЪЕМ ДИСЦИПЛИНЫ

с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

Очная форма обучения

№ п/п	Наименование разделов и тем	Всего	Аудиторные занятия			Сам. раб.
			ЛК	ПЗ	ЛБ	
6 семестр						
	Раздел 1. Основы языка Python для анализа данных и моделирования	180	18	36	18	108
1	Тема 1. Предметно-ориентированное программирование и особенности Python для Data Science	20	2	4	2	12
2	Тема 2. Основные структуры данных: списки, кортежи, словари, множества	20	2	4	2	12
3	Тема 3. Массивы и матрицы: NumPy как основа научных вычислений	20	2	4	2	12
4	Тема 4. Табличные данные: Pandas DataFrame и Series	20	2	4	2	12
5	Тема 5. Основные программные конструкции Python: циклы, условия, функции	20	2	4	2	12
6	Тема 6. Математические вычисления и моделирование с SciPy и SymPy	20	2	4	2	12
7	Тема 7. Объектно-ориентированное программирование в Python	20	2	4	2	12
8	Тема 8. Ввод/вывод данных: работа с файлами (CSV, JSON, Excel, базы данных)	20	2	4	2	12
9	Тема 9. Работа со строками и регулярными выражениями	20	2	4	2	12
	<i>Зачет</i>					

	<i>Итого за 6 семестр</i>	<i>180</i>	<i>18</i>	<i>36</i>	<i>18</i>	<i>108</i>
7 семестр						
	Раздел 2. Дополнительные возможности Python для моделирования и анализа	108	10	20	20	47,2
10	Тема 10. Анализ данных: агрегация, фильтрация, группировка в Pandas	20	2	4	4	5
11	Тема 11. Визуализация данных: Matplotlib, Seaborn, Plotly	20	2	4	4	5
12	Тема 12. Отладка и профилирование кода: pdb, cProfile, logging	20	2	4	4	10
13	Тема 13. Параллелизм и асинхронность: multiprocessing, threading, asyncio	20	2	4	4	10
14	Тема 14. Установка и использование пакетов для моделирования: Scikit-learn, Statsmodels, TensorFlow	28,5	2	4	4	17,2
	<i>Контроль</i>	<i>9</i>				
	<i>Экзамен</i>	<i>0,3</i>				
	<i>Курсовой проект</i>	<i>0,5</i>				
	<i>Иные формы работы</i>					
	<i>Итого за 6 семестр</i>	<i>108</i>	<i>10</i>	<i>20</i>	<i>20</i>	<i>47,2</i>
	в т.ч. практическая подготовка	8				
	ИТОГО:	288	28	56	38	155,2

Очно-заочная форма обучения не реализуется (не реализуется)

Заочная форма обучения не реализуется (не реализуется)

III. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Текущая аттестация проводится в форме контрольной работы, теста.

Типовой вариант контрольной работы в 6 семестре

Вариант 1. Провести регрессионный анализ экзаменационных оценок с использованием scikit-learn.

Вариант 2. Осуществить прогнозирование временных рядов с дискретными значениями statsmodels.

Вариант 3. Провести два полных сеанса отладки с использованием pdb и logging.

Типовой вариант тестов в 7 семестре

A1. Какой результат выведет данная программа?

```
a = "Hello, World!"
```

```
print(len(a))
```

1) 2; 2) 10; 3) 1; 4) 13

A2. Что выведет программа?

```
E = 'no'
```

```
print(E * 5)
```

1) no5; 2) Ошибка; 3) false; 4) nononono

A3. Каким будет результат выполнения кода?

```
a = [1, 2, 3]
```

```
if a[1] < 3:
```

```
    print(a[a[0]-1])
```

```
else:
```

```
    print(a[0])
```

1) 1; 2) 2; 3) 3; 4) Ошибка.

A4. Табличная структура данных в Python, где каждый столбец может иметь свой тип, называется:

1) DataFrame; 2) матрица; 3) ndarray; 4) словарь

A5. Сколько раз выполнится цикл?

```
for i in range(1, 6):
```

```
    print(i)
```

1) 4; 2) 5; 3) 6; 4) Ошибка

A6. Какой тип данных создаст код?

```
x = [True, False, True]
```

```
type(x[0])
```

1) int; 2) str; 3) bool; 4) object

Примерная тематика рефератов

1. История и эволюция Python в Data Science.
2. Сравнение структур данных: list, numpy.array, pandas.DataFrame.
3. Функциональное программирование в Python: map, filter, lambda.
4. Работа с Big Data: Dask vs Pandas.
5. Визуализация данных: от Matplotlib до Plotly Dash.
6. Статистический анализ с помощью SciPy и Statsmodels.
7. Основы машинного обучения на Scikit-learn.
8. NLP в Python: NLTK, spaCy, transformers.
9. Создание пакетов Python: структура, setup.py, PyPI.
10. Использование Python в биоинформатике (Biopython).
11. Интерактивные веб-приложения: Flask, Streamlit, Dash.
12. Параллельные вычисления: multiprocessing, joblib, Dask.
13. Интеграция Python с C/C++ (Cython, ctypes) и R (rpy2).
14. Геоданные: GeoPandas, Folium, Plotly Maps.
15. Автоматизация отчетов: Jupyter Notebook, nbconvert, Voilà.
16. Анализ временных рядов: Prophet, ARIMA в Python.
17. Финансовый анализ: pandas-datareader, yfinance, backtesting.
18. Сравнение Python с R, SAS, SPSS.

19. Работа с API, базами данных (SQLAlchemy), веб-скрапинг (BeautifulSoup, Scrapy).

Промежуточная аттестация обучающихся осуществляется в форме зачета, экзамена, курсового проекта, с использованием следующих оценочных материалов: перечень вопросов к зачету, перечень вопросов к экзамену, перечень тем курсового проекта.

Вопросы к зачету (6 семестр, очная форма обучения)

1. Основные структуры данных в Python. Примеры.
2. Массивы NumPy: создание, индексация, операции.
3. Создание и использование списков и словарей.
4. Работа с DataFrame в Pandas. Отличие от массивов NumPy.
5. Категориальные данные в Pandas (pd.Categorical).
6. Фильтрация данных: .loc, .query(), булева индексация.
7. Обработка пропущенных значений: NaN, None.
8. Управляющие конструкции: if-else, for, while, list comprehensions.
9. Создание функций. Аргументы по умолчанию, *args, **kwargs.
10. Функции apply, map, transform в Pandas.
11. Математические операции: NumPy ufuncs, scipy.stats.
12. ООП в Python: классы, методы, наследование.
13. Чтение/запись файлов: CSV, JSON, Excel, pickle.
14. Регулярные выражения: модуль re.
15. Замыкания в Python.
16. Отладка: модуль pdb, print-отладка, logging.
17. Построение графиков: matplotlib.pyplot.
18. Установка пакетов: pip, conda, виртуальные окружения.
19. Интеграция Python с другими языками (Cython, subprocess).

Вопросы к экзамену (7 семестр, очная форма обучения)

1. Механизм замыканий и их применение.
2. Рекурсия и хвостовая рекурсия.
3. Глобальные и нелокальные переменные (global, nonlocal).
4. Векторизация в NumPy и ее преимущества.
5. Параллелизм: multiprocessing, concurrent.futures.
6. Создание пользовательских операторов (не поддерживается — объяснить почему).
7. Оптимизация памяти: генераторы, slots, del.
8. 3D-графика: matplotlib 3D, Plotly.
9. Различия между классами и dataclasses / Pydantic.
10. Ускорение кода: Numba, Cython.

11. Имитационное моделирование: SimPy, дискретные события.
12. Профилирование: cProfile, line_profiler, memory_profiler.
13. Построение субплотов: plt.subplots(), gridspec.
14. Декораторы: создание и применение.
15. Анонимные функции: lambda.
16. Асинхронность: async/await, asyncio.
17. Кэширование: functools.lru_cache.
18. Сохранение состояния: pickle, joblib, dill.
19. Работа с большими файлами: chunking, итераторы.
20. Создание REST API: Flask, FastAPI.

Примерные темы курсового проекта (7 семестр, очная форма обучения)

1. Анализ временных рядов: прогнозирование с ARIMA/Prophet.
2. Визуализация статистических данных с интерактивными дашбордами (Plotly Dash).
3. Оптимизация кода для Big Data с Dask.
4. Модель машинного обучения: классификация с Scikit-learn + оценка качества.
5. Многомерный регрессионный анализ и визуализация.
6. Имитационное моделирование очередей (SimPy).
7. Функциональный подход к обработке данных: map/filter/reduce.
8. Анализ текстов: тематическое моделирование (LDA, NMF).
9. Интерактивные отчеты: Jupyter Notebook → Voilà или Streamlit.
10. Параллельная обработка изображений/данных с multiprocessing.
11. Разработка пакета Python для статистического анализа.
12. Финансовый анализ: волатильность, корреляция активов, бэктестинг.
13. Визуализация геоданных: кластеризация точек на карте.
14. Медицинская аналитика: предсказание диагноза по данным (kaggle-датасеты).
15. Оптимизация памяти и скорости: профилирование и применение Numba/Cython.

IV. ПЕРЕЧЕНЬ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

4.1. Основная литература

1. Чернышев, С. А. Основы программирования на Python : учебник для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 349 с. — (Высшее образование). — ISBN 978-5-534-17139-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/567821> (дата обращения: 21.04.2025).

4.2. Дополнительная литература

1. Федоров, Д. Ю. Программирование на python : учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 187 с. — (Высшее образование). — ISBN 978-5-534-19666-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/556864> (дата обращения: 21.04.2025).

2. Гниденко, И. Г. Технологии и методы программирования : учебник для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 241 с. — (Высшее образование). — ISBN 978-5-534-18130-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/581329> (дата обращения: 21.04.2025).

V. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

№ пп	Ссылка на информационный ресурс	Наименование разработки в электронной форме	Доступность
1.	http://edu.ru/	Российское образование: Федеральный портал. Включает ссылки на порталы и сайты образовательных учреждений; государственные образовательные стандарты; нормативные документы; каталог экскурсий и обучающих программ.	Свободный доступ
2.	https://infourok.ru/	Инфоурок: образовательный интернет-проект России. Включает: конспекты уроков, презентации, тесты, видеоуроки и другие материалы по предметам школьной программы.	Свободный доступ

VI. СОВРЕМЕННЫЕ ПРОФЕССИОНАЛЬНЫЕ БАЗЫ ДАННЫХ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

1.	http://www.biblioclub.ru	Электронно-библиотечная система (ЭБС) Университетская библиотека онлайн	Регистрация через любой университетский компьютер. В дальнейшем предоставляется неограниченный индивидуальный доступ из любой точки, в которой имеется доступ к сети Интернет
2.	www.elibrary.ru	Российский информационный портал в области науки, технологии,	Свободный доступ

		медицины и образования	
3.	www.consultant.ru	Российская компьютерная справочно-правовая система	Свободный доступ
4.	www.garant.ru	Информационно-правовой портал	Свободный доступ

VII. ЛИЦЕНЗИОННОЕ И СВОБОДНО РАСПРОСТРАНЯЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

При реализации учебной дисциплины применяется следующее лицензионное и свободно распространяемое программное обеспечение:

- Microsoft Windows;
- Microsoft Office;
- LibreOffice;

VIII. ОБОРУДОВАНИЕ И ТЕХНИЧЕСКИЕ СРЕДСТВА ОБУЧЕНИЯ, НЕОБХОДИМЫЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Учебные занятия проводятся в аудиториях, укомплектованных специализированной мебелью, в том числе стационарными или переносными техническими средствами обучения (проектор, экран, компьютер/ноутбук).

Лабораторные занятия, групповые и индивидуальные консультации, текущая и промежуточная аттестации проводятся в специализированных классах, оснащенных автоматизированными рабочими местами с компьютерами.

Самостоятельная работа проводится в кабинетах, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.