

Компонент ОПОП 09.04.01 Информатика и вычислительная техника
Направленность (профиль) Компьютерный анализ и интерпретация данных.
Data Science.

Б1.О.08
ШИФР дисциплины

РАБОЧАЯ ПРОГРАММА

Дисциплины
(модуля)

Основы Python

Разработчик:

Золотов О.В.

ФИО

доцент

должность

канд. физ.-мат. наук


ученая степень,
звание

Утверждено на заседании кафедры
Информационных технологий

наименование кафедры

протокол № 6 от 01.02.2024

Заведующий кафедрой ИТ



подпись

ФИО

Ляш О.И.

Мурманск
2024

Пояснительная записка

Объем дисциплины 4 з.е.

1. Результаты обучения по дисциплине (модулю), соотнесенные с индикаторами достижения компетенций, установленными образовательной программой

Компетенции	Индикаторы достижения компетенций	Результаты обучения по дисциплине (модулю)
<p>ОПК-2. Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</p>	<p>ИД-1_{опк-2} Понимает принципы и применяет методы разработки алгоритмов и программных средств</p> <p>ИД-2_{опк-2} Использует современные интеллектуальные технологии при разработке алгоритмов и программных средств</p> <p>ИД-2_{опк-3} Решает профессиональные задачи путем разработки оригинальных алгоритмов и программных средств</p>	<p>Знать:</p> <ul style="list-style-type: none"> - основные принципы и методы разработки алгоритмов и программных продуктов на языке программирования высокого уровня (ЯВУ) Python 3; - современные методы и технологии, применяемые при разработке алгоритмов и программных средств на ЯВУ Python 3; - существующие в Python-сообществе лучшие практики, применяемые при решении профессиональных задач на ЯВУ Python 3; - основы построения и функционирования информационных и автоматизированных систем. <p>Уметь:</p> <ul style="list-style-type: none"> - применять принципы и методы разработки алгоритмов и программных продуктов на ЯВУ Python 3; - применять современные методы и технологии, используемые при разработке алгоритмов и программных средств на ЯВУ Python 3; - решать профессиональные задачи путем разработки оригинальных алгоритмов и программных средств на ЯВУ Python 3; - применять в профессиональной деятельности принципы построения и функционирования информационных и автоматизированных систем на ЯВУ Python 3.
<p>ОПК-5. Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем</p>	<p>ИД-1_{опк-5} Понимает принципы построения и функционирования информационных и автоматизированных систем</p> <p>ИД-2_{опк-5} Разрабатывает и модернизирует программное обеспечение информационных и автоматизированных систем</p> <p>ИД-3_{опк-5} Разрабатывает и модернизирует аппаратное обеспечение информационных и автоматизированных систем</p>	<p>Владеть:</p> <ul style="list-style-type: none"> - навыком применения принципов и методов разработки алгоритмов и программных продуктов на ЯВУ Python 3; - навыком применения современных методов и технологий, используемых при разработке алгоритмов и программных средств на ЯВУ Python 3; - навыком решения профессиональных задач путем разработки оригинальных алгоритмов и программных средств на ЯВУ Python 3; - навыком применения принципов построения и функционирования элементов информационных и автоматизированных систем.

2. Содержание дисциплины (модуля)

Тема 1. *Введение в Python3. Среда и окружение разработки.*

Интерпретируемые языки программирования, их особенности. Краткая история развития Python, области применения, поддерживаемые операционные системы. Виртуальная машина Python3. Проблема глобальной блокировки (Проблема GIL). Доступные реализации интерпретатора Python3. Получение, развертывание и настройка дистрибутива Python3 для целевой операционной системы. Менеджер пакетов pip и репозиторий пакетов PyPI. Виртуальное окружение (на примере venv). Интерактивный режим работы интерпретатора Python. Оболочка IPython. Получение справки, перечня доступных полей и методов классов в Python. Интерактивная среда разработки PyCharm Community, ее основные возможности, установка и настройка.

Тема 2. *Основные конструкции ЯВУ Python3.*

Особенности написания программ на Python: требования к форматированию и оформлению (PEP8). Простая программа на Python3. Кодировка. Комментарии. Идентификаторы и ключевые слова. Литералы. Операторы. Разделители. Понятие типа данных в Python. Переменные. Допустимые операции. Множественное присваивание / позиционное присваивание. Понятие модуля. «Простые» встроенные типы данных Python: целые, с плавающей точкой и комплексные; булевы (логический), диапазоны (range vs xrange). Тип None. Строки в Python: «простые» операции над строками, срезы. Краткое ознакомление с консольным вводом-выводом. Управление потоком исполнения: ветвления и циклы. Функции zip(), enumerate(). Списки, операции со списками, особенности представления в памяти. Срезы. Списковые выражения (списковые включения) и генераторы списков. Кортежи. Создание кортежей при помощи генераторов. Упаковка и распаковка переменных. Ассоциативные контейнеры (словари). Словарные выражения (включения) и генераторы словарей. Множества. Множественные выражения и генераторы множеств.

Тема 3. *Структура программы и функции ЯВУ Python3.*

Главная точка входа `__main__`. Оператор pass. Оператор assert. Импорт пакетов, классов, функций. Определение пользовательских функций. Области видимости функций и переменных (global и nonlocal). Позиционные аргументы функций. Именованные аргументы. Значения по умолчанию. Пустой список как значение по умолчанию. Функции с произвольным количеством аргументов. Передача функции как аргумента. Лямбда-выражения. Функции sum, min, max, sorted, all, any, map, reduce, filter, apply. Декораторы.

Тема 4. *Объектно-ориентированное программирование на ЯВУ Python3.*

Объекты и классы. Использование готовых классов. Исключения. Обработка исключений. Файловый ввод-вывод. Обращение к стандартным потокам ввода-вывода как к файловым потокам. Менеджеры контекста. Итераторы. Работа с датой и временем. Коллекции Python. Типы bytes, bytearray, memoryview. Знакомство с пакетами sys, os, copy, math, cmath, random. Определение собственных классов. Поля и методы классов, их область видимости. Управление доступом к полям и методам. Свойства. Наследование. Суперклассы и подклассы. Множественное наследование. Статические классы и статические методы.

Тема 5. *Тестирование и подготовка к распространению программного*

обеспечения, реализованного на ЯВУ Python.

Возможности фреймворка для юнит-тестирования pytest. Организация тестового кода. Пакеты в Python3. Форматы 'egg' («яйцо») и 'wheel' («колесо»). Создание и распространение собственного пакета в Python3 (на примере setuptools); стандартная структура и назначение файлов и папок пакета. Работа с пакетом в редактируемом режиме. Спецификация и управление зависимостями пакетов. Включение произвольных бинарных файлов в пакет. Вложенные пакеты. Публичные репозитории пакетов PyPI и Test.PyPI. Загрузка и размещение пакетов в PyPI и Test.PyPI.

3. Перечень учебно-методического обеспечения дисциплины (модуля)

- мультимедийные презентационные материалы по дисциплине (модулю) представлены в электронном курсе в ЭИОС МАУ;
- методические указания к выполнению лабораторных работ представлены в электронном курсе в ЭИОС МАУ;
- методические материалы для обучающихся по освоению дисциплины (модуля) представлены на официальном сайте МАУ в разделе «Информация по образовательным программам, в том числе адаптированным».

4. Фонд оценочных средств по дисциплине (модулю)

Является отдельным компонентом образовательной программы, разработан в форме отдельного документа, включает в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины (модуля);
- задания текущего контроля;
- задания промежуточной аттестации;
- задания внутренней оценки качества образования.

5. Перечень основной и дополнительной учебной литературы (печатные издания, электронные учебные издания и (или) ресурсы электронно-библиотечных систем)

Основная литература:

1. Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин. – Ростов-на-Дону : Южный федеральный университет, 2022. – 242 с. – Текст : электронный. – URL: <https://elibrary.ru/item.asp?id=49349781>
2. Кондратьева, В. А. Основы программирования на языке Python : учебное пособие / В. А. Кондратьева. – Москва : Московский городской педагогический университет, 2021. – 92 с. – Текст : электронный. – URL: <https://elibrary.ru/item.asp?id=46438002>
3. Салимьянов, И. Т. Программирование на языке Python / И. Т. Салимьянов, В. Т. Якупова. – Казань : Редакционно-издательский центр «Школа», 2019. – 90 с. – Текст : электронный. – URL: <https://www.elibrary.ru/item.asp?id=41589506>
4. Сысоева, М. В. Программирование для «нормальных» с нуля на языке Python. Учебник. В 2 частях. Часть 1 / М. В. Сысоева, И. В. Сысоев. – Москва : МАКС Пресс, 2018. – 176 с. – Текст : электронный. – URL: <https://www.elibrary.ru/item.asp?id=34882407>

Дополнительная литература:

1. The Python Tutorial. – Text : electronic // Python Documentation : [site]. – [S. I.], 2022. – URL: <https://docs.python.org/3/tutorial/>

2. The Python Language Reference. – Text : electronic // Python Documentation : [site]. – [S. I.], 2022. – URL: <https://docs.python.org/3/reference/>
3. The Python Standard Library. – Text : electronic // Python Documentation : [site]. – [S. I.], 2022. – URL: <https://docs.python.org/3/library/>
4. Smith K. D., PEP 318 – Decorators for Functions and Methods / K. D. Smith, J. J. Jewett, S. Montanaro, A. Baxter. – Text : electronic // Python Enhancement Proposals : [site]. – [S. I.], 2003. – URL: <https://peps.python.org/pep-0318/>
5. pytest: helps you write better programs. – Text : electronic // pytest.org : [site]. – S. I.], 2015. – URL: <https://docs.pytest.org/>
6. pytest-dependency - Manage dependencies of tests. – Text : electronic // Read the Docs : [site]. – [S. I.], 2022. – URL: <https://pytest-dependency.readthedocs.io/en/latest/index.html>
7. Installing Packages. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/tutorials/installing-packages/>
8. Installing packages using pip and virtual environments. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/installing-using-pip-and-virtual-environments/>
9. Building and Distributing Packages with Setuptools // Setuptools : a library designed to facilitate packaging Python projects : [site]. – [2024]. – URL: <https://setuptools.pypa.io/en/latest/userguide/>
10. Packaging Python Projects. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/tutorials/packaging-projects/>
11. Packaging and distributing projects. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/distributing-packages-using-setuptools/>
12. Including files in source distributions with MANIFEST.in. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/using-manifest-in/>
13. Packaging namespace packages. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/packaging-namespace-packages/>
14. Using TestPyPI. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/using-testpypi/>
15. Making a PyPI-friendly README. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/making-a-pypi-friendly-readme/>
16. Tool recommendations. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/guides/tool-recommendations/>
17. An Overview of Packaging for Python. – Text : electronic // PyPA : [site]. – [S. I.], 2022. – URL: <https://packaging.python.org/en/latest/overview/>

6. Профессиональные базы данных и информационные справочные системы

1. eLIBRARY.RU : научная электронная библиотека. – Москва, 2000- . – URL: <https://www.elibrary.ru/> . – Режим доступа: научная электронная библиотека eLIBRARY.RU, после регистрации.
2. Kaggle : AI & ML community : site. – 2024. – URL: <https://www.kaggle.com/>
3. КонсультантПлюс : справочная правовая система : сайт. – Москва, 1997- . – URL: <https://www.consultant.ru>

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства

1. Средство запуска виртуальных машин Virtual Box или Hyper-V.

2. Свободное программное обеспечение Rocky Linux v. 9.4+.
3. Свободное программное обеспечение интерпретатор Python3.
4. Свободное программное обеспечение – менеджер пакетов pip.
5. Свободное программное обеспечение PyCharm Community – интегрированная среда разработки;
6. Браузер, поставляемый вместе с операционной системой (либо свободное программное обеспечение Mozilla FireFox или Google Chrome).

8. Обеспечение освоения дисциплины лиц с инвалидностью и ОВЗ

Обучающиеся из числа инвалидов и лиц с ОВЗ обеспечиваются печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

9. Материально-техническое обеспечение дисциплины (модуля) представлено в приложении к ОПОП «Материально-технические условия реализации образовательной программы» и включает:

- учебные аудитории для проведения учебных занятий, предусмотренных программой магистратуры, оснащенные оборудованием и техническими средствами обучения;

- помещения для самостоятельной работы обучающихся, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к электронной информационно-образовательной среде МАУ.

Допускается замена оборудования его виртуальными аналогами.

10. Распределение трудоемкости по видам учебной деятельности

Таблица 1 - Распределение трудоемкости

Вид учебной деятельности	Распределение трудоемкости дисциплины (модуля) по формам обучения											
	Очная				Очно-заочная				Заочная			
	Семестр			Всего часов	Семестр			Всего часов	Семестр/Курс			Всего часов
	1	-	-		-	-	-		-	-	-	
Лекции	20	-	-	20	-	-	-	-	-	-	-	-
Практические занятия	-	-	-	-	-	-	-	-	-	-	-	-
Лабораторные работы	24	-	-	24	-	-	-	-	-	-	-	-
Самостоятельная работа	64	-	-	64	-	-	-	-	-	-	-	-
Подготовка к промежуточной аттестации ¹	36	-	-	36	-	-	-	-	-	-	-	-
Всего часов по дисциплине	144	-	-	144	-	-	-	-	-	-	-	-
/ из них в форме практической подготовки ²	34	-	-	34	-	-	-	-	-	-	-	-

Формы промежуточной аттестации и текущего контроля

Экзамен	+	-	-	1	-	-	-	-	-	-	-	-
Зачет/зачет оценкой ^с	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-	-/-
Курсовая работа (проект)	-	-	-	-	-	-	-	-	-	-	-	-
Количество расчетно-графических работ	1	-	-	1	-	-	-	-	-	-	-	-
Количество контрольных работ	-	-	-	-	-	-	-	-	-	-	-	-
Количество рефератов	-	-	-	-	-	-	-	-	-	-	-	-
Количество эссе	-	-	-	-	-	-	-	-	-	-	-	-

Перечень лабораторных работ по формам обучения

1 Для экзамена очной и очно-заочной формы обучения - 36 часов, для экзамена заочной формы обучения - 9 часов, для зачета заочной формы обучения - 4 часа.

2 Организуется при реализации учебных дисциплин (модулей) путем проведения практических занятий, лабораторных работ и иных аналогичных видов учебной деятельности, предусматривающих участие обучающихся в выполнении **отдельных элементов работ, связанных с будущей профессиональной деятельностью.**

Практическая подготовка может включать в себя отдельные занятия лекционного типа, которые предусматривают передачу учебной информации обучающимся, необходимой для последующего выполнения работ, связанных с будущей профессиональной деятельностью.

№ п\п	Темы лабораторных работ
1	2
	Очная форма
1	Организация рабочего места Python-разработчика: развертывание и настройка Python-окружения, средств разработки
2	Изучение основных конструкций Python3 (I)
3	Изучение основных конструкций Python3 (II)
4	Структура программы и функции в Python3 (I)
5	Структура программы и функции в Python3 (II)
6	Объектно-ориентированное программирование в Python3 (I)
7	Объектно-ориентированное программирование в Python3 (II)
8	Объектно-ориентированное программирование в Python3 (III)
9	Тестирование Python-кода (I)
10	Тестирование Python-кода (II)
11	Создание и подготовка к распространению Python-пакетов (I)
12	Создание и подготовка к распространению Python-пакетов (II)
	Заочная форма
-	Заочная форма обучения не реализуется

Перечень практических занятий по формам обучения

№ п\п	Темы практических занятий
1	2
	Очная форма
-	Практические занятия не предусмотрены учебным планом
	Заочная форма
-	Заочная форма обучения не реализуется

Перечень примерных тем курсовой работы /курсового проекта

№ п\п	Темы курсовой работы / проекта
1	2
-	Курсовая работа /проект не предусмотрены учебным планом