

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»**

**УТВЕРЖДЕНО**

**Проректор по учебной работе и  
довузовской подготовке**

**А.А. Воронов**

	<b>Рабочая программа дисциплины (модуля)</b>
<b>по дисциплине:</b>	Язык Python
<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Системная и синтетическая биология Физтех-школа Биологической и Медицинской Физики кафедра информатики и вычислительной математики
<b>курс:</b>	2
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 3 (осенний) - Зачет

Аудиторных часов: 30 всего, в том числе:

лекции: 30 час.

семинары: 0 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 15 час.

Всего часов: 45, всего зач. ед.: 1

Программу составил: Т.Ф. Хирьянов, старший преподаватель

Программа обсуждена на заседании кафедры информатики и вычислительной математики 27.06.2023

## Аннотация

Курс направлен на изучение возможностей языка Python 3 и среды Jupyter для агрегации данных и разведочного анализа данных.

В частности, происходит изучение инструментария библиотек Matplotlib, NumPy, Pandas.

### 1. Цели и задачи

#### Цель дисциплины

Изучение языка Python, среды Jupyter и библиотек для обработки данных и визуализации результатов лабораторных работ по естественнонаучным дисциплинам.

#### Задачи дисциплины

- знакомство с современным инструментарием обработки данных для языка Python;
- знакомство с возможностями библиотеки Numpy;
- знакомство с типами Series и DataFrame библиотеки Pandas;
- приобретение умения пользоваться многомерными массивами Numpy;
- приобретение умения в программе на Python считывать и записывать табличные файлы в формате CSV;
- приобретение умения фильтровать данные таблиц Pandas по сложным условиям, преобразовывать и сливать их друг с другом;
- знакомство с возможностями визуализации графиков и диаграмм в Matplotlib и Seaborn;
- формирование навыков работы в среде Jupyter включая использование горячих клавиш, языка разметки Markdown и записи формул в нотации LaTeX.

### 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-4 Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах)	УК-4.2 Использует современные информационно-коммуникативные средства для коммуникации
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.3 Знает основные требования информационной безопасности
	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области
	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ПК-3 Способен выбирать и применять подходящее оборудование, инструменты и методы исследований для решения задач в избранной предметной области	ПК-3.2 Знает области и критерии применимости используемых теоретических подходов и умение оценивать точность приближенных аналитических методов вычислений
	ПК-3.3 Умеет производить оценку точности численных методов, используемых на ЭВМ, вычислительной сложности используемых алгоритмов и объема требуемых вычислительных ресурсов

### 3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- основы синтаксиса языка Python;
- возможности научных библиотек Python по анализу и визуализации данных.

уметь:

- работать в средах Jupyter Notebook и JupyterLab;
- создавать программы на языке Python в том числе в формате Jupyter Notebook;
- использовать библиотеки Pandas и Numpy для анализа экспериментальных данных;
- пользоваться разметкой Markdown для создания ячеек-пояснений в Jupyter;
- пользоваться LaTeX для написания формул;
- визуализировать полученные при анализе результаты.

владеть:

- навыками работы в среде Jupyter включая использование горячих клавиш, языка разметки Markdown и записи формул в нотации LaTeX;
- инструментарием языка Python и научных библиотек для анализа данных на практике.

#### 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

##### 4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Основы Python	12			5
2	Использование Jupyter Notebook и JupyterLab	2			1
3	Основы Numpy	6			3
4	Основы Pandas	6			3
5	Визуализация данных и зависимостей в Matplotlib и Seaborn	4			3
Итого часов		30			15
Подготовка к экзамену		0 час.			
Общая трудоёмкость		45 час., 1 зач.ед.			

##### 4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 3 (Осенний)

###### 1. Основы Python

Преимущества и недостатки языка Python 3

Дзен Python. Antigravity

Python2 и Python3

Ресурсы для обучения Python: stepic.com, checkio.org, pythontutor.com

Концепция присваивания в Python

Переменные, значения и их типы. Понятие о динамической типизации.

Обмен двух переменных значениями.

Кортежи и их использование.

Кортежи переменных. Обмен значений.

Арифметические операции. Возведение в степень, деление нацело.

«Hello, World!» на Python

Цикл while. Инструкции управления циклом.

Позиционные системы счисления

Литералы чисел в Python

Оператор if. Каскадная условная конструкция elif.  
Логические операции в Python.  
Основы алгебры логики  
ASCII и Unicode.  
Тип str. Длина строки len(s). Неизменяемость строки.  
Срезы строк.  
Методы строк find, count, replace, startswith, endswith.  
Наивный поиск подстроки в строке.  
Приведение строки к числу с указанием системы счисления.  
Тип list. Изменяемость списка.  
Ссылочная модель данных в Python. Операторы == и is. Копирование объектов.  
Алгоритм обращения массива.  
Алгоритм циклического сдвига в массиве.  
Срезы списков. Присваивание в срез. Методы списка.  
Стандартные функции len, max, min, sum.  
Список строк. Методы split и join для строки.  
Тип tuple как замороженный list.  
Тип set. Множества и работа с ними.  
Тип dict. Словарь (ассоциативный массив) и операции с ним.  
Dict comprehensions: генерация множеств и словарей.  
Частотный анализ для строк.  
Генераторы, yield.  
Подключение модулей инструкцией import  
Модуль math  
Модуль random  
Запись арифметических выражений в выражения на Python.  
Создание функции в Python.  
Полиморфизм в Python. Duck typing.  
Значения параметров по умолчанию.  
Именованные параметры.

## 2. Использование Jupyter Notebook и JupyterLab

Установка и запуск Jupyter Notebook и JupyterLab.  
Принципы использования Jupyter. Когда он подходит, а когда нет.  
Создание ячеек и их порядок.  
Синтаксис Markdown текстовых ячеек.  
Управление подсветкой синтаксиса вставок кода на разных языках программирования.  
Вставка изображений и графиков.  
Синтаксис ввода формул LaTeX в ячейках Jupyter.

## 3. Основы NumPy

Установка и подключение NumPy.  
Массивы ndarray: отличие от списков list и стандартных массивов array.  
Простые типы данных NumPy. Фиксированное число бит для чисел.  
Способы создания массивов NumPy.  
Векторные операции с массивами.  
Срезы массивов NumPy.  
Выборка элементов по логическому критерию.  
Матричные операции в NumPy.  
Линейная алгебра в NumPy.

## 4. Основы Pandas

Установка и подключение Pandas.

Типы Series и DataFrame для работы с сериями и таблицами данных.

Индексация серий и фреймов. Локаторы loc и iloc. Срезы по индексам.

Векторные операции с сериями. Логические операции &, | и особенности их приоритета.

Выборка строк по логическому условию. Метод query.

Статистика данных в таблице. Перцентили, медиана, среднее, отклонение. Гистограммы.

Функции агрегации данных. Группировка по категориальным параметрам.

## 5. Визуализация данных и зависимостей в Matplotlib и Seaborn

Установка и подключение Matplotlib и Seaborn.

Типы графиков, диаграмм, гистограмм. Адекватность их применения для визуализации данных.

Управление цветами, видами линий, подписями на графиках.

Трёхмерные графики.

Анимация графиков.

## 5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория, оснащенная мультимедиа проектором и экраном для чтения лекций.

## 6. Перечень рекомендуемой литературы

### Основная литература

1. Python и анализ данных, Первичная обработка данных с применением pandas, NumPy и IPython / У. Маккини. — Москва, ДМК Пресс, 2020.— URL: <https://e.lanbook.com/book/131721> (дата обращения: 26.01.2021). - Полный текст (Режим доступа : из сети МФТИ / Удаленный доступ)

### Дополнительная литература

1. Python и машинное обучение [Текст], крайне необходимое издание по новейшей предсказательной аналитике для более глубокого понимания методологии машинного обучения/С. Рашка, -М., ДМК Пресс, 2017
2. Программирование на Python 3, подробное руководство/М. Саммерфилд,-СПб, Символ-Плюс, 2020

## 7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

1. <http://www.intuit.ru>,
2. <http://cs.mipt.ru>

## 8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

На лекциях используются мультимедийные технологии, включая демонстрацию презентаций.

Для контроля и коррекции знаний обучающихся используются автоматизированное компьютерное тестирование на основе Ejudge или CMS Moodle.

## 9. Методические указания для обучающихся по освоению дисциплины (модуля)

Данный курс не предполагает аудиторных лабораторных или семинарских занятий, поэтому успешное освоение курса требует от студента высокой ответственности и самоорганизации.

Изложение материала происходит преимущественно на лекциях, но важная часть материала предназначена для самостоятельного изучения в ходе выполнения домашних работ.

Самостоятельная работа включает в себя:

- проработку лекционного материала;
- еженедельное изучение учебно-методического текста, прилагающегося к лекции;
- еженедельное решение задач домашних работ;
- чтение рекомендованной литературы.

#### # Учёт, контроль и оценка знаний студентов

Посещаемость лекций не отмечается, но каждая домашняя работа завязана на материал прошедшей лекции, что делает посещение лекций насущной необходимостью в течение семестра. Успеваемость отслеживается по результатам и своевременности сдачи домашних работ. В середине семестра выявляются отстающие студенты с передачей докладных в деканат.

Зачёт проводится в форме дистанционной контрольной работы — онлайн-контеста. Студенты, выполнившие все домашние работы своевременно и на высокий балл, имеют право на получение зачёта без итоговой контрольной. Если преподаватель сомневается в самостоятельности выполнения домашних работ или итоговой работы, студент вызывается на устный зачёт по материалу лекций. Устный ответ практически исключает списывание, показывает владение базовой терминологией предмета, а также позволяет проверить знание концепций и подходов к анализу данных.

Обращаем внимание, что домашняя работа предполагает самостоятельное выполнение, а не умение пользоваться помощью товарищей или задавать вопросы длинным языковым моделям!

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

**по направлению:** Прикладные математика и физика  
**профиль подготовки:** Системная и синтетическая биология  
Физтех-школа Биологической и Медицинской Физики  
кафедра информатики и вычислительной математики  
**курс:** 2  
**квалификация:** бакалавр

Семестр, формы промежуточной аттестации: 3 (осенний) - Зачет

**Разработчик:** Т.Ф. Хирьянов, старший преподаватель

## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-4 Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах)	УК-4.2 Использует современные информационно-коммуникативные средства для коммуникации
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.3 Знает основные требования информационной безопасности
	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области
	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ПК-3 Способен выбирать и применять подходящее оборудование, инструменты и методы исследований для решения задач в избранной предметной области	ПК-3.2 Знает области и критерии применимости используемых теоретических подходов и умение оценивать точность приближенных аналитических методов вычислений
	ПК-3.3 Умеет производить оценку точности численных методов, используемых на ЭВМ, вычислительной сложности используемых алгоритмов и объема требуемых вычислительных ресурсов

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Язык Python» обучающийся должен:

### знать:

- основы синтаксиса языка Python;
- возможности научных библиотек Python по анализу и визуализации данных.

### уметь:

- работать в средах Jupyter Notebook и JupyterLab;
- создавать программы на языке Python в том числе в формате Jupyter Notebook;
- использовать библиотеки Pandas и Numpy для анализа экспериментальных данных;
- пользоваться разметкой Markdown для создания ячеек-пояснений в Jupyter;
- пользоваться LaTeX для написания формул;
- визуализировать полученные при анализе результаты.

### владеть:

- навыками работы в среде Jupyter включая использование горячих клавиш, языка разметки Markdown и записи формул в нотации LaTeX;
- инструментарием языка Python и научных библиотек для анализа данных на практике.

## 3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

Примеры типовых задач для текущего контроля включают задачи создания различной сложности классов с использованием магических методов и итераторов:

1. Класс трехмерных векторов.
2. Класс квадратных матриц.
3. Класс обработки текстовой информации с итератором по словам по частоте встречаемости в тексте.
4. Класс для осуществления классификатора набора числовых данных.
5. Очередь без использования list.
6. Стек без использования list.



7. Именованный массив, без использования dict.
8. Создать иеррархию классов, включающую параллелограмм, квадрат и ромб.
9. Класс двоичных деревьев поиска.
10. Класс heap.

От всех студентов требуется умение создавать стандартные генераторы map, range, zip, enumerate и декораторы.

#### **4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся**

Перечень контрольных вопросов для дифференцированного зачета:

1. Ссылочная модель данных, параметры функция по умолчанию, функции с неизвестным числом параметров.
2. Парадигмы ООП, SOLID принципы.
3. Создание класса в Python, инстанцирование, декоратор @staticmethod.
4. Магические методы классов.
5. Абстрактный класс, библиотека abc.
6. Генераторы, оператор yield.
7. Итераторы в Python.
8. Сопроцессы.
9. Асинхронное программирование, asyncio.
10. Регулярные выражения, библиотека re.
11. Библиотека threading, GIL.
12. Библиотека subprocess.
13. Обмен данными между процессами, pipe.
14. Работа с фалами Python.
15. Декораторы в Python.
16. Сериализация и десериализация с использованием pickle

#### **Критерии оценивания**

Зачёт проводится в форме дистанционной контрольной работы — онлайн-контеста.

Студенты, выполнившие все домашние работы своевременно и на высокий балл, имеют право на получение оценки "зачтено" без итоговой контрольной.

Если преподаватель сомневается в самостоятельности выполнения домашних работ или итоговой работы, студент вызывается на устный зачёт по материалу лекций.

Устный ответ практически исключает списывание, показывает владение базовой терминологией предмета, а также позволяет проверить знание концепций и подходов к анализу данных.

"Не зачтено" ставится в том случае, если студент в устном ответе продемонстрировал отсутствие каких-либо знаний языка Python или концепций и подходов анализа данных.

В против ном случае, ставиться оценка "зачтено"

#### **5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

Время проведения дифференцированного зачёта составляет 30 минут на одного обучающегося.

Во время подготовки к ответу обучающиеся не могут пользоваться литературой, печатными материалами, рукописными записями, а также электронными средствами (сотовыми телефонами, планшетами, умными часами и т.п.).