**Межфакультетский курс**

**Основы программирования и анализа данных на Python.**

**Аннотация.**

Курс посвящен основам программирования на языке Python. Слушатели курса познакомятся с основными конструкциями языка Python и парадигмами программирования, а также с инструментами для анализа и визуализации данных.

Основная цель курса - это изучение основ работы с данными с помощью языка Python.

Темы курса:

* синтаксис языка Python;
* коллекции «золотого фонда» Python;
* визуализация данных;
* библиотеки NumPy и Pandas;
* извлечение и получение данных;
* парадигмы программирования.

Для кого курс?

1. Для тех, кто не изучал программирование, но настроен серьезно и готов посвящать много времени и сил освоению данного направления.
2. Для тех, кто изучал другой (не Python) язык программирования.

В случае если вы никогда не сталкивались с программированием, данный курс потребует некоторых усилий для его освоения. Если вы пока к этому не готовы, то мы рекомендуем сначала пройти курс “Введение в программирование на основе языка Python”, а потом вернуться к данному курсу.

Видеотрансляции занятий будут доступны всем желающим. Студенты, записавшиеся на курс, смогут выполнять домашние задания с автоматической проверкой и получать обратную связь от преподавателей.

Данный курс является составляющей частью серии курсов по искусственному интеллекту в Московском государственном университете. Рекомендован к прослушиванию параллельно с курсом по математике для анализа данных в онлайн-формате. Продолжением данного курса являются курсы по машинному обучению, нейронным сетям, инженерным вычислениям. Цель этой серии курсов — предоставление студентам актуальных и современных знаний в области искусственного интеллекта и в различных областях науки о данных.

На курсе “Основы программирования и анализа данных на Python” будут рассмотрены инструменты, необходимые для дальнейшего обучения на курсах по тематике искусственного интеллекта. Курс относится к МФК в области ИИ.

Telegram-канал, в котором будут публиковаться объявления по курсам: <https://t.me/msumfk>

Материалы курса: <https://teach-in.ru/course/python-programming-and-data-analysis-basics/about>

**Объем курса.**

24 академических часа (12 лекций).

**Количество мест.**

750 человек.

**Формат проведения.**

Онлайн (с демонстрацией лектора, презентации и программного кода).

**Форма итоговой аттестации.**

Зачет по результатам выполненных домашних работ.

# **Программа курса “Основы программирования и анализа данных на Python”**

1. Вводная лекция
   1. Обзор возможностей Python
   2. Алгоритмы (с помощью библиотеки “черепашка”)
2. Текстовые данные
   1. Функции print (sep, end) и input
   2. Сложение и умножение строк
   3. Ошибки, где искать помощь
   4. Типы данных
   5. Память программы
   6. Строки и методы строк (lower, upper, title, replace) , len
   7. for по строке
   8. Срезы строк
   9. Считывание из файлов
3. Итерируемые объекты
   1. Переменные
   2. Функции
   3. Множественное присваивание, распаковка
   4. Списки, создание списков, конструирование списков (join, split)
   5. Срезы списков
   6. Генераторы списков
   7. map
   8. zip
   9. itertools
   10. Запись в файл
4. Числовые данные
   1. Двоичная система счисления
   2. Дробные числа (Fraction, Decimal, float)
   3. Переменные
   4. Обмен переменных значениями
   5. Подсчет ситуаций
   6. Суммирование чисел
   7. while
   8. range
   9. Условная конструкция if / elif/ else
5. Сортировка данных
   1. Ссылочная модель данных
   2. Изменение элементов списка
   3. Копирование списков
   4. Присваивание в срез списка
   5. Удаление и добавление элементов списка
   6. Сортировка списка
   7. Список списков
   8. Вложенная генерация
6. Частотный анализ и группировка данных
   1. Частотный анализ (метод count, вопрос эффективности, альтернативная реализация)
   2. Множество set
      1. Итерируемый объект
      2. Принадлежность множеству
      3. Операции со множествами
      4. Генерация множества, добавление и удаление элементов
   3. Словари dict
      1. Добавление и удаление элементов
      2. Частотный анализ со словарем
   4. Модуль collections
7. Построение графиков в Matplotlib
   1. Дробные числа, числа с плавающей точкой и ошибки вычислений
   2. Модуль math. Виды округлений: floor, ceil, round
   3. Стандартные математические функции: тригонометрические, экспонента, логарифм
   4. Установка библиотек NumPy и Matplotlib
   5. Построение графика функции
   6. Описание def пользовательской функции
   7. Вывод экспериментальных данных с ошибками
   8. Зачем нужны массивы np.array вместо списков list?
   9. np.arange и математические функции в NumPy
   10. Срезы массивов в NumPy
8. Библиотека NumPy
   1. Генерация случайных чисел
   2. Статистика выборки: среднее арифметическое, среднеквадратическое отклонение, медиана, максимум, минимум
   3. Случайные распределения
   4. Откуда возникают матрицы и какие операции с ними возможны
9. Библиотека Pandas
   1. Логический тип bool, условные выражения, функции all и any
   2. Pandas Series
   3. Индексы и срезы
   4. Фильтрация
10. Библиотека Pandas, часть 2
11. Сбор данных
    1. Регулярные выражения
    2. Веб-парсинг
    3. Библиотеки request и beautifulsoap
12. Парадигмы программирования
    1. Повторное использование программного кода
    2. Структурная парадигмы (декомпозиция на функции)
    3. Модульная парадигма
    4. Объектно-ориентированное программирование в Python

**Вопросы (задачи) к зачету по курсу “Основы программирования и анализа данных на Python”**

Задачи нужно выполнить на языке программирования Python и сдать в систему автоматической проверки Яндекс.Контест.

1. Задана строка состоящая из символов. Необходимо посчитать их количество .
2. Измените порядок символов в строке s на обратный.
3. Найдите подстроку s' заключенную в двойных кавычках в строке s.
4. Писатель Яша купил себе новую пишущую машинку и сразу бросился писать свой новый роман. Закончив пятнадцатую главу и решив оценить свое творение, Яша заметил, что в пишущей машинке был очень необычный заводской брак: все слова она писала задом-наперёд, хотя и в нужном порядке. У Яши больше нет сил, чтобы переписывать роман с самого начала.  
   Помогите опечаленному Яше справиться с его бедой: напишите программу, которая развернет все слова в строке, не меняя их порядка.  
   Дано предложение в одной строке. Вывести предложение, содержащее слова в исходном порядке, но каждое слово записано задом-наперёд
5. Буква **ё** в рукописных и печатных текстах по различным причинам часто заменяется на 'е'. Эту особенность традиции письменности в русском языке можно отслеживать и использовать для установления оригинальности текстов, для поиска грамматических ошибок, для других целей, связанных с анализом текстов. Напишите программу, которая будет искать строки в текстовом файле, в котором есть буква 'ё' и выводить их на экран  
   Дан файл 'input.txt' . При помощи функции .filter() и lambda-функции вывести на экран все строки файла, содержащие строчную букву **ё**.
6. Нотариус Зоя помогает юридически неграмотным гражданам оформлять их жалобы и запросы в государственные службы и коммерческие компании. У Зои трудная работа: многие граждане совсем не умеют не только составлять юридически верные тексты, но ещё и оформлять свои мысли в сколько-нибудь связной форме.  
   Однажды клиентом Зои стал подросток Вадим. Вадим очень любил копаться в компьютерах и прочей технике и очень не любил уроки литературы, которые активно прогуливал. Вадим принес Зое папку жалоб на разные компании, в которых он заказывал компоненты компьютеров - они часто доставлялись в неудовлетворительном состоянии.  
   Прочитав первую жалобу, Зоя обомлела: слова-паразиты встречались в тексте Вадима буквально через слово.  
   Вот пример одной из записок Вадима: " Мне типа привезли этот товар ну в вообще плохом совсем состоянии. Плата там разбита прямо. Я вот хочу типа вернуть это, деньги. "  
   Зоя поняла, что исправить все записки Вадима вручную - задача непосильная. Помогите Зое, исключив каждое второе слово из текстов Вадима.  
   Дан файл input.txt. Вывести все строки этого файла, но в каждой строке уберите каждое второе слово.
7. Дан текстовый файл input.txt, содержащий некоторое нечетное количество строк. В первом строке файла записано число N. В следующих N строках содержатся имена студентов, а в следующих N строках (то есть в строках, имеющих номера N+2 - 2N+2) — их даты рождения. При помощи функции zip соедините построчно информацию из этих файлов и запишите в файл output.txt, при этом разделяйте имена и даты рождения при помощи спец.символа табуляции '\t'
8. Классу 8 "Б" было поручено организовать школьные дежурства в новом учебном году. Ученики должны дежурить по трое, и перед классным руководителем была поставлена задача организовать тройки дежурных, составив план дежурств.  
   8 "Б" - очень дружный класс. Когда пришло время назначать план школьных дежурств, каждый из учеников 8 "Б" напрочь отказался принимать в этом участие, пока ему не пообещают, что в течении года он подежурит с каждым из своих друзей.  
   Классный руководитель 8 "Б" и учитель истории Мария Ивановна - замечательный человек и прекрасный педагог. Она решила пойти навстречу своим ученикам. Мария Ивановна сразу поняла, что ей предстоит решить сложную комбинаторную задачу и не мешкая приступила к опросу своих учеников.

* Вася, скажи мне, кто твои друзья, с которыми ты хочешь дежурить?
* Мария Ивановна, весь наш класс - мои друзья!
* Молодец, Вася, то есть ты хочешь подежурить с каждым из одноклассников. А ты, Петя?
* Мария Ивановна, и я, как Вася, дружу со всеми ребятами!

К окончанию опроса выяснилось, что все ребята дружат со всеми. Мария Ивановна сразу сообразила, что ей предстоит организовать троек, где N - количество ребят в классе. Помогите Марии Ивановне, написав программу, составляющую всевозможные тройки дежурных.  
Дан текстовый файл studygroup.txt, содержащий имена школьников 8 "Б". При помощи itertools.combinations вывести все способы назначения трёх дежурных

1. Перевести число из двоичной системы счисления в десятичную.
2. Посчитать при помощи класса Fraction библиотеки fractions следующее выражение:
3. Посчитать сумму последовательность целых чисел, вводимых пользователем с клавиатуры, которая завершается нулём.
4. Дана геометрическая прогрессия: .  
     
   Найти номер *n* первого члена геометрической прогрессии, который превышает заданное пользователем число *a*.
5. На лекции вы изучили, как рассчитывать таблицы для погашения кредита в банке.  
   По аналогии создайте программу для расчета банковского вклада на заданный срок. Вклад без ежемесячного пополнения, но с ежемесячным начислением процентов и капитализацией (начислением процентов на проценты). Результат выведите в командую строку.
6. На лекции вы изучили, как рассчитывать таблицы для погашения кредита в банке.  
   По аналогии создайте программу для расчета банковского вклада на заданный срок. Вклад без ежемесячного пополнения, но с ежемесячным начислением процентов и капитализацией (начислением процентов на проценты). Результат сохраните в файле output.csv.
7. Для игры в американский футбол собрались 22 мальчишки с разным весом. Их вес (у всех разный) и их имена (тоже у всех разные) даны в файле weights.txt. Требуется сделать две команды следующим образом: отсортировать всех ребят по убыванию веса, а затем каждого второго по порядку отобрать в 1-ю команду, а оставшихся — во 2-ю команду. Вывести список команды №1 в первые 11 строк файла team.txt, а список команды №2 в последние 11 строк файла team.txt. В списках команд указывать и имена, и веса.
8. Журналист обнаружил в архиве ранее не публиковавшуюся шифровку Эдгара По poe\_unpublished.txt. Чтобы не просто представить её публике, а совершить сенсацию, он хочет расшифровать её и для этого нанял вас. Вы разобрались, что в тексте намеренно перепутаны строки, а в каждой строке перепутаны слова. Для применения алгоритма расшифрования нужно упорядочить строки по возрастанию количества слов в каждой строке (слова разделяются пробелами), а внутри каждой строки слова нужно упорядочить по количеству букв в них. Результат нужно сохранить в файл poe\_decode\_attempt.txt
9. Биолог провёл исследование устойчивости нескольких видов бактерий стрептококка к нескольким антибиотикам, и получил численные оценки, характеризующие эффективность действия каждого препарата. Его данные хранятся в формате текстовой таблицы дробных чисел (с плавающей точкой) med\_research.csv, при этом по горизонтали изменяются виды бактерий, а по вертикали — виды антибиотиков. Однако для отчёта требуется, чтобы они шли по горизонтали, а виды бактерий — по вертикали. Биолог узнал, что эта операция называется в математике транспонированием, но он не знает как её сделать автоматически. Если вы не поможете, то ему предстоит провести ночь, вручную переставляя числа в огромном листе. В файле хранятся только сами числа. Список видов бактерий и список антибиотиков вас не интересуют, биолог вставит их сам.
10. Руководитель хочет узнать какой из двух его заместителей А или Б больше времени тратит на звонки по телефону. Для этого он установил шпионское ПО на их смартфоны, и получил файл the\_calls.txt, в котором для каждого звонка по телефону указано: время его совершения (дата, пробел, время в формате YYYY.MM.DD hh:mm:ss, длительность звонка в секундах (целое число), буква А или Б, а также номер телефона, на который был совершён звонок (без пробелов, но со значком + и скобками). Поля в строке разделяются знаком табуляции.  
    Руководителю более подозрительны звонки, длившиеся дольше всего, поэтому строки следует отсортировать в обратном порядке по длительности звонка.

Распределите строки из файла на два упорядоченных указанным образом списка, и запишите их в файл calls.txt: Сначала группу звонков А, затем - группу звонков В

1. Создайте список и при помощи присваивания в срез заполните его следующим образом:  
   1. [0, 0, 1, 0, 0, 2, 0, 0, 3, 0, 0, 4, 0, 0, 5]

2. [0, 0, 5, 0, 0, 4, 0, 0, 3, 0, 0, 2, 0, 0, 1]

3. [0, 13, 2, 11, 4, 9, 6, 7, 8, 5, 10, 3, 12, 1]

4. [0, 10, 20, 30, 40, 1, 2, 3, 4, 400, 300, 200, 100, 5, 6, 7, 8, 9]

5. [0, 1, 2, 3, 4, 7, 8, 9]

1. Провести частотный анализ числа и найти наиболее часто встречающуюся цифру.
2. В определенном зоопарке находятся животные, каждому из которых присвоены следующее: ID, вид, пол, кличка, дата рождения и дата поступления в зоопарк. Исходя из этих данных, вывести все виды животных, которые есть в данном зоопарке.
3. В определенном зоопарке находятся животные, каждому из которых присвоены следующее: ID, вид, пол, кличка, дата рождения и дата поступления в зоопарк. Исходя из этих данных, вывести количество животных для каждого вида в порядке убывания.
4. В определенном зоопарке находятся животные, каждому из которых присвоены следующее: ID, вид, пол, кличка, дата рождения и дата поступления в зоопарк. Исходя из этих данных, вывести ID всех животных для каждого вида.
5. В роддом одного провинциального города прибыл областной журналист. Он рассказал, что в роддоме ожидается рождение 100 000 жителя города, что является большим событием и должно быть освещено в прессе. Помимо данных самого младенца, журналист хочет узнать данные о всех младенцах, рожденных в этом роддоме, чтобы убедиться, что именно 100 000 житель города родится самым (ну или хотя бы одним из самых) физически превосходным. Редакция журнала решила, что самым важным параметром для новорожденных мальчиков является рост, в то время как для девочек - это вес. Помогите журналисту разобраться в записях о новорожденных младенцах провинциального роддома.  
   В файле input.csv даны обезличенные характеристики множества новорождённых: вес, рост, пол. Определите, входит ли 100000 житель города в топ-10% по всем новорожденным
6. Пекарь Иван работает в кондитерской. Иван с самого утра готовит много выпечки, чтобы в течение дня продать ее. Иногда бывает, что Иван готовит слишком много выпечки, и она пропадает, а иногда - слишком мало, и клиенты остаются недовольны. В обоих случаях Иван несет убытки. Он обратился к другу-математику Григорию, чтобы тот помог ему. Проанализировав данные о посещаемости пекарни Ивана за последний год, Григорий выяснил, что функция зависимости количества посетителей пекарни от дня месяца задается следующим выражением:   
   Помогите Ивану посчитать, сколько клиентов у него будет 18 числа месяца, реализовав функцию при помощи инструментов python. Ответ округлите до целого при помощи int()
7. Физик Леонид придумал, как решить все проблемы современной физики. Он придумал новую формулу, но для того, чтобы воспользоваться ей ему нужно сложить число, порядка 1e-12 и число порядка 1e-55. Попробовав сделать это, удивленный Леонид обнаружил, что его программа выдает ему какой-то неправильный результат:  
   1*e*−12+1*e*−55=1*e*−12  
   Леонид спросил совета у своего друга-программиста, и тот рассказал ему, что все дело в том, что вещественные числа в представлении ЭВМ - очень большое, но все же дискретное множество, и что пытаясь сложить два очень маленьких числа, Леонид вызывает т.н. underflow problem. Расстроенный Леонид решил, что может пренебречь точностью вычислений, но ему нужно знать, какой минимальный порядок числа он может прибавить к числу 1e-12, чтобы программа смогла исполниться корректно, то есть 1e-12 + b != 1e-12. Помогите Леониду найти такое число  
     
   Совет: Вы можете задать число типа float при помощи вызова функции float(str), где str будет иметь вид '1e-n'
8. В процессе археологических раскопок было найдено много документов. Откапывая все новые и новые рукописи, археологи датировали их и складывали в одну стопку. У одного из археологов родилась новая гипотеза о причинах поражения Французов в битве 1415 года при Азенкуре. Однако, для проверки гипотезы необходимо найти документ 1415 года. Вам дан список дат, которые приписаны к каждому документу. Даты упорядочены по возрастанию. При помощи бинарного поиска найдите индекс соответствующего документа. Гарантируется, что документ 1415 года есть в списке.
9. Менеджер крупной столичной фирмы Алексей готовит отчет о прибыльности перспективного проекта за последние дни. Для отчета о прибыли фирмы Алексей проанализировал много данных и построил графики дневной прибыли. К удивлению Алексея, проект оказался настолько перспективным и прибыльным, что график получается практически линейным.То есть, если за первый день работы проекта прибыль была 5 у.е., за последний - 15 у.е., между ними прошло 10 дней, то прибыль за второй составит 6 у.е., за третий - 7 у.е. и тд. Проект работает всю неделю, но внезапно оказалось, что участники проекта очень не любят понедельники, и их продуктивность по понедельникам (то есть каждый 7 день) снижается в 3 раза по сравнению с ожидаемой, а в пятницу - наоборот: участники проекты чувствуют душевный подъем и работают в 2 раза активнее. Из-за этого прибыль по этим дням изменяется пропорционально продуктивности  
   Пример: ожидаемая прибыль проекта  
   -пн: 6 у.е. -вт: 8 у.е. -ср: 10 у.е. -чт: 12 у.е. -пт: 14 у.е. -сб: 16 у.е. -вс: 18 у.е. -пн: 20 у.е. -вт: ... и т.д.  
   С поправкой на продуктивность:  
   -пн: 2 у.е. -вт: 8 у.е. -ср: 10 у.е. -чт: 12 у.е. -пт: 28 у.е. -сб: 16 у.е. -вс: 18 у.е. -пн: 6.666 у.е. -вт: ... и т.д.  
   Вам дана информация о прибыли в первый день запуска проекта, последний день запуска проекта и количестве дней. Считаем, что понедельник - всегда первый день запуска проекта. При помощи np.linspace и присваиваний в срез - массивы ожидаемой прибыли проекта с поправкой на продуктивность участников
10. В один приятный вечер три подруги-модницы встретились, чтобы обсудить последние новости моды. Каждая из них пришла в платье своего цвета: Вилена пришла в Зеленом платье, Марианна - в салатовом, а Снежанна - в платье цвета лесной листвы. Они принялись спорить, чей цвет моднее в этом сезоне. Для того, чтобы решить спор, подруги откопали 150 выпусков журнала "Мода модных модниц", и стали считать, сколько раз известные звезды появлялись в том или ином цвете. Считав массив, содержащий в себе информацию об этих подсчетах, определите, какой же цвет самый модный
11. В спортивном клубе "Две чугунные гири" есть два типа клиентов:

* одни из них всегда делают примерно одинаковое количество подходов, планомерно развивая свои физические способности
* другие же часто ленятся и делают существенно меньше, чем нужно, а затем их мучает совесть, и они начинают делать существенно больше.

Внимательные тренеры ведут скрупулезные и подробные записи о каждом посетителе клуба. Отличить одних от других можно при помощи подсчета стандартного отклонения по статистике количества сделанных подходов за последние дни. Нормальным разбросом вокруг среднего значения подходов для последовательных посетителей считается +-4 подхода.  
По записям за последние 30 дней определите кого в клубе больше - клиентов 1 или 2 типа. Гарантируется, что одних из них больше

1. Программист Иван - очень общительный человек. Поэтому в рабочее время он любит общаться с друзьями в социальных сетях. Недавно программист Иван узнал, что его начальники логируют время, проведенное сотрудниками в социальных сетях и в тот же момент понял, что его карьера висит на волоске. Он выяснил у своего друга и коллеги, как можно получить доступ к логам, и в каком формате логируются данные. Оказалось, что данные логируются в виде двумерного массива размерностью 30 x 12, где каждый элемент массива содержит информацию о том, сколько времени сотрудник провел в социальных сетях в i-й рабочий день j-го месяца, в часах. Программист Иван знает, что начальство поймет, что что-то не так, если Иван просто обнулит все свои логи. Иван внимательно с ними ознакомился и узнал, что наибольшее время в социальных сетях он проводил по четным числам нечетных месяцев, и по четным числам - нечетных. Помогите Ивану занизить логи, уменьшив ровно в 2 раза элементы массива, имеющие один четный и один нечетный индекс (Ради удобства начальство Ивана считает, что каждый месяц состоит ровно из 30 дней).  
   Вам дан csv-файл с логами, его нужно считать и при помощи операций с массивами (мы рекомендуем Вам использовать присваивание в срез) изменить его так, как сказано в условии задачи. А затем сохранить в csv-файл.
2. В выручай-комнате Гарри Поттер и его соратники тренировались сражаться на волшебных палочках, устраивая дуэли. Результаты дуэлей тщательно записывались в волшебный список. К концу месяца друзья составляли списки лучших волшебников. Однажды оказалось, что Гарри стал не лучшим волшебником, и руководство дуэльного клуба в лице Гарри и его ближайших друзей, решило, что это плохо для имиджа клуба, поэтому список нужно немного скорректировать следующим образом: В каждом месяце Гарри должен выигрывать в полтора раза больше матчей, чем ребята выигрывали в среднем. Этот список волшебный и изменить его можно только с помощью змеиного языка Python, которым Гарри, как известно, отлично владеет. Напишите для Гарри заклинание, которое он должен произнести, чтобы имидж клуба не пострадал.  
     
   Вам дан csv-файл с количеством побед каждого из участников клуба (их 20) за каждый из 30 дней месяца. Присваиванием в срез замените значение в первой строке (конечно, эта строка принадлежит Гарри) на среднее значение побед по этому дню, умноженное на 1.5
3. В математической статистике есть много интересных парадоксов и загадок. Одним из самых известных и интересных примеров является т.н. ошибка выжившего. Этот парадокс имеет много интерпретаций: например, есть известная история о том, как во время второй мировой войны американские инженеры принимали решение о том, каким образом можно было бы укрепить части самолетов. У них были ресурсы для укрепления лишь одной части: двигателя, лобового стекла, крыльев или хвоста самолета. Большинство самолетов возвращались с повреждениями в районе хвоста и крыльев - логичным решение кажется укреплять одну из этих частей. Но все не так просто: в воздушном бою снаряды попадают по самолетам равномерно, поэтому факт того, что большинство самолетов возвращается с повреждениями в этих областях, говорит лишь о некритичности попаданий в эти области, поскольку самолетам все же хватает ресурсов для возвращения на базу. Тогда укреплять нужно область с минимальным количеством попаданий.  
     
   Вам предлагается отследить ошибку выжившего в следующем формате:  
     
   Вам дан csv-файл, в котором содержится информация об N вернувшихся на базу самолетах. Мы условно разделили самолет на 10 областей, в двумерном массиве размерностью N x 10 приведено количество попаданий в каждую область из этих областей у вернувшегося самолета. Посчитайте среднее количеств попаданий для каждой области по всем самолетам и примите решение о том, какую из 10 областей Вы бы стали укреплять.