

# Знакомство с языком Python (семинары)

## Урок 1. Ввод-Вывод, операторы ветвления

### Задача 1: Сумма цифр трехзначного числа

Найдите сумму цифр трехзначного числа  $n$ .

Результат сохраните в переменную `res`.

**Пример:**

$n = 123 \rightarrow res = 6 \ (1 + 2 + 3)$

$n = 100 \rightarrow res = 1 \ (1 + 0 + 0)$

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значение
3  `n`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `n` для проверки
7  """
8
9
10 # n = 123
11
12 # Введите ваше решение ниже
13
14 res =
```

### Задача 2: Бумажные журавлики

Петя, Катя и Сережа делают из бумаги журавликов. Вместе они сделали  $n$  журавликов.

Сколько журавликов сделал каждый ребенок, если известно, что Петя и Сережа сделали одинаковое количество журавликов, а Катя сделала в два раза больше журавликов, чем Петя и Сережа вместе?

Выведите кортеж из количества журавликов, сделанных Петей, Катей и Сережей.

**Пример:**

$n = 6 \rightarrow (1, 4, 1)$

$n = 24 \rightarrow (4, 16, 4)$

$n = 60 \rightarrow (10, 40, 10)$

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значение
3  `n`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `n` для проверки
7  """
8
9
10 # n = 6
11
12 # Введите ваше решение ниже
```

### Задача 3: Счастливый билет

Вы пользуетесь общественным транспортом? Вероятно, вы расплачивались за проезд и получали билет с номером.

Счастливым билетом называют такой билет с шестизначным номером, где сумма первых трех цифр равна сумме последних трех.

Т.е. билет с номером 385916 – счастливый, т.к.  $3+8+5=9+1+6$ .

Вам требуется написать программу, которая проверяет счастливость билета с номером  $n$  и выводит на экран yes или no.

**Пример:**

$n = 385916 \rightarrow \text{yes}$

$n = 123456 \rightarrow \text{no}$

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значение
3  `n`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `n` для проверки
7  """
8
9  # n = 385916
10
11 # Введите ваше решение ниже
```

### Задача 4: Деление шоколадки

Инструкция по использованию платформы

Определите, можно ли от шоколадки размером  $a \times b$  долек отломить  $c$  долек, если разрешается сделать один разлом по прямой между дольками (то есть разломить шоколадку на два прямоугольника).

Выведите yes или no соответственно.

**Пример:**

$a, b, c = 3, 2, 4 \rightarrow \text{yes}$

$a, b, c = 3, 2, 1 \rightarrow \text{no}$

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значения
3  `a`, `b`, `c`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `a`, `b`, `c` для проверки"""
7
8  # a = 3
9  # b = 2
10 # c = 4
11
12 # Введите ваше решение ниже
```

## Урок 2. Циклы (for, while)

**Задача 10:** На столе лежат  $n$  монеток. Некоторые из них лежат вверх решкой, а некоторые – гербом. Определите минимальное число монеток, которые нужно перевернуть, чтобы все монетки были повернуты вверх одной и той же стороной. Выведите минимальное количество монет, которые нужно перевернуть

**Задача 12:** Петя и Катя – брат и сестра. Петя – студент, а Катя – школьница. Петя помогает Кате по математике. Он задумывает два натуральных числа  $X$  и  $Y$  ( $X, Y \leq 1000$ ), а Катя должна их отгадать. Для этого Петя делает две подсказки. Он называет сумму этих чисел  $S$  и их произведение  $P$ . Помогите Кате отгадать задуманные Петей числа.

**Задача 14:** Требуется вывести все целые степени двойки (т.е. числа вида  $2^k$ ), не превосходящие числа  $N$ .

**задача 1 сложная необязательная** Посчитать сумму цифр любого целого или вещественного числа. Через строку решать нельзя.

**задача Де моргана необязательная**

Напишите программу для проверки истинности утверждения  $\neg(X \vee Y \vee Z) = \neg X \wedge \neg Y \wedge \neg Z$  для всех значений предикат.

теперь надо проверить ее практически

в цикле 100 раз прогоняем

каждый раз генерируем случайное количество предикат от 3 до 15

и конечно со случайным булевым значением

и засекаем общее время выполнения программы

юзаем библиотеки random и time

предикаты НЕ ЗАДАЕМ как целое число!

**задача 3 необязательная**

Валентина прогуляла лекцию по математике.

Преподаватель решил подшутить над нерадивой студенткой и

попросил ее на практическом занятии перечислить все положительные делители некоторых целых чисел.

Для несложных примеров студентка быстро нашла решения (для числа 6 это: 1, 2, 3, 6; а для числа 16 это: 1, 2, 4, 8, 16), но этим все не закончилось.

На домашнее задание ей дали варианты посложнее: 23436, 190187200, 380457890232.

Решить такое вручную, как вы понимаете, практически нереально.

Вот Валентина и обратилась к вам за помощью.

Помогите ей (при помощи функции `all_divisors(number)`, которую напишете сами).

Постарайтесь найти самое оптимальное решение.

Результат представьте в виде списка (не забудьте отсортировать по возрастанию).

### Урок 3. Списки и словари

Для решения данного домашнего задания вам необходимо воспользоваться сервисом автоматической проверки написанного кода.

Для того, чтобы успешно выполнить задание, необходимо перейти по каждой из представленных ссылок и решить все предложенные задачи. Будьте внимательны, количество попыток отправки кода на проверку ограничено! Вам дано 5 попыток на каждую задачу.

Прикреплять полученные решения не требуется. Итоговая оценка домашнего задания появится автоматически на платформе после решения всех задач.

Полученная оценка не повлияет на получение итогового документа об обучении.

**Задача 3.1:** Требуется вычислить, сколько раз встречается некоторое число  $k$  в массиве `list_1`.

Найдите количество и выведите его.

**Пример:**

```
list_1 = [1, 2, 3, 4, 5]
```

```
k = 3
```

```
# 1
```

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значения
3  `list_1` и `k`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `list_1` и `k` для проверки
7  """
8
9  # list_1 = [1, 2, 3, 4, 5]
10 # k = 3
11
12 # Введите ваше решение ниже
```

**Задача 3.2:** Требуется найти в массиве `list_1` самый близкий по величине элемент к заданному числу `k` и вывести его.

**Пример:**

```
list_1 = [1, 2, 3, 4, 5]
```

```
k = 6
```

```
# 5
```

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значения
3  `list_1` и `k`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `list_1` и `k` для проверки
7  """
8
9  # list_1 = [1, 2, 3, 4, 5]
10 # k = 6
11
12 # Введите ваше решение ниже
```

**Задача 3.3:** В настольной игре **Скрабл (Scrabble)** каждая буква имеет определенную ценность.

В случае с **английским** алфавитом очки распределяются так:

- A, E, I, O, U, L, N, S, T, R – 1 очко;
- D, G – 2 очка;
- B, C, M, P – 3 очка;
- F, H, V, W, Y – 4 очка;
- K – 5 очков;
- J, X – 8 очков;
- Q, Z – 10 очков.

А **русские** буквы оцениваются так:

- А, В, Е, И, Н, О, Р, С, Т – 1 очко;
- Д, К, Л, М, П, У – 2 очка;
- Б, Г, Ё, Ъ, Я – 3 очка;
- Й, Ы – 4 очка;
- Ж, З, Х, Ц, Ч – 5 очков;
- Ш, Э, Ю – 8 очков;

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значение
3  `k`
4
5  При отправке решения на Проверку закомментируйте эту строку обратно - система
6  автоматически подставит разные значения `k` для проверки
7  """
8
9  # k = 'ноутбук'
10
11 # Введите ваше решение ниже
12
```

## Урок 4. Словари, множества и профилирование

**Задача 22:** Даны два неупорядоченных набора целых чисел (может быть, с повторениями). Выдать без повторений в порядке возрастания все те числа, которые встречаются в обоих наборах.

Пользователь вводит 2 числа.  $n$  — кол-во элементов первого множества.  $m$  — кол-во элементов второго множества. Затем пользователь вводит сами элементы множеств.

**Задача 24:** В фермерском хозяйстве в Карелии выращивают чернику. Она растёт на круглой грядке, причём кусты высажены только по окружности. Таким образом, у каждого куста есть ровно два соседних. Всего на грядке растёт  $N$  кустов.

Эти кусты обладают разной урожайностью, поэтому ко времени сбора на них выросло различное число ягод — на  $i$ -ом кусте выросло  $a[i]$  ягод.

В этом фермерском хозяйстве внедрена система автоматического сбора черники. Эта система состоит из управляющего модуля и нескольких собирающих модулей. Собирающий модуль за один заход, находясь непосредственно перед некоторым кустом, собирает ягоды с этого куста и с двух соседних с ним.

Напишите программу для нахождения максимального числа ягод, которое может собрать за один заход собирающий модуль, находясь перед некоторым кустом заданной во входном списке урожайности грядки.

**задача 1 необязательная.** Напишите программу, которая получает целое число и возвращает его двоичное, восьмеричное строковое представление.

Функции `bin` и `oct` используйте для проверки своего результата.

\*Дополнительно

Попробуйте избежать дублирования кода в преобразованиях к разным системам счисления

Избегайте магических чисел

Добавьте аннотацию типов где это возможно

Используйте функции

**задача 2 необязательная.**

Даны два многочлена, которые вводит пользователь.

Задача - сформировать многочлен, содержащий сумму многочленов.

Степени многочленов могут быть разные.

например на входе  $2x^2 + 4x + 5 = 0$  и  $5x^3 - 3x^2 - 12 = 0$

на выходе будет  $5x^3 - x^2 + 4x - 7 = 0$

можно использовать модуль `re`

## Урок 5. Рекурсия и алгоритмы

Для решения данного домашнего задания вам необходимо воспользоваться сервисом автоматической проверки написанного кода.

Для того, чтобы успешно выполнить задание, необходимо перейти по каждой из представленных ссылок и решить все предложенные задачи. Будьте внимательны, количество попыток отправки кода на проверку ограничено! Вам дано 5 попыток на каждую задачу.

Прикреплять полученные решения не требуется. Итоговая оценка домашнего задания появится автоматически на платформе после решения всех задач.

Полученная оценка не повлияет на получение итогового документа об обучении.

**Задача 5.1:** Напишите функцию  $f$ , которая на вход принимает два числа  $a$  и  $b$ , и возводит число  $a$  в целую степень  $b$  с помощью рекурсии.

Функция не должна ничего выводить, только возвращать значение.

**Пример:**

$a = 3; b = 5 \rightarrow 243 \ (3^5)$

$a = 2; b = 3 \rightarrow 8$

```
1  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значения
2  переменных `a` и `b` и вызвать функцию
3
4  При отправке решения на Проверку закомментируйте эти строки обратно - система
5  автоматически подставит разные значения `a` и `b` и сама вызовет функцию для проверки
6  """
7
8  # Введите ваше решение ниже
9
10 # a = 3
11 # b = 5
12 # print(f(a, b))
```



**Задача 5.2:** Напишите рекурсивную функцию `sum(a, b)`, возвращающую сумму двух целых неотрицательных чисел. Из всех арифметических операций допускаются только `+` и `-`. Также нельзя использовать циклы.

Функция не должна ничего выводить, только возвращать значение.

**Пример:**

`sum(2, 2)`

# 4

```
1  """
2  При отправке кода на Выполнение раскомментируйте строку ниже, чтобы задать значения
3  переменных `a` и `b` и вызвать функцию
4
5  При отправке решения на Проверку закомментируйте эти строки обратно - система
6  автоматически подставит разные значения `a` и `b` и сама вызовет функцию для проверки
7  """
8
9  # Введите ваше решение ниже
10
11 # a = 3
12 # b = 5
13 # print(sum(a, b))
```

**Задача 5.3: необязательная.** Напишите рекурсивную программу вычисления арифметического выражения заданного строкой. Используйте операции `+`, `-`, `/`, `*`. приоритет операций стандартный.

\*Пример:\*

`2+2 => 4;`

`1+2*3 => 7;`

`1-2*3 => -5;`

- Добавьте возможность использования скобок, меняющих приоритет операций.

\*Пример:\*

`1+2*3 => 7;`

`(1+2)*3 => 9;`

Тут может помочь библиотека `re`

## Оформлять всё через функции! это обязательное условие

**Задача 30:** Заполните массив элементами арифметической прогрессии. Её первый элемент, разность и количество элементов нужно ввести с клавиатуры. Формула для получения n-го члена прогрессии:  $a_n = a_1 + (n-1) * d$ . Каждое число вводится с новой строки.

**Задача 32:** Определить индексы элементов массива (списка), значения которых принадлежат заданному диапазону (т.е. не меньше заданного минимума и не больше заданного максимума). Список можно задавать рандомно  
На входе : [ 1, 5, 88, 100, 2, -4]

33

200

Ответ: [2, 3]

**Задача XO необязательная.**

Сделайте игру крестики - нолики, человек должен играть с ботом, поле 3\*3.

Конечно, бот не должен ходить на занятые поля

Если есть желание, то можете наделить бота псевдоинтеллектом, чтоб он ходил как можно ближе к своим занятым клеткам

После каждого хода на экран должна выводиться текущая обстановка на поле.

Например,

```
|   | x |  
|   | o |  
|   | o |
```

При ходе пользователя у надо спрашивать номер строки и столбца, куда он хочет сделать ход

**Задача FOOTBALL необязательная**

Напишите программу, которая принимает на стандартный вход список игр футбольных команд с результатом матча и выводит на стандартный вывод сводную таблицу результатов всех матчей.

За победу команде начисляется 3 очка, за поражение — 0, за ничью — 1.

Формат ввода следующий:

В первой строке указано целое число  $nn$  — количество завершённых игр.

После этого идет  $nn$  строк, в которых записаны результаты игры в следующем формате:

Перваякоманда;Забитопервойкомандой;Втораякоманда;Забитоввторойкомандой

Вывод программы необходимо оформить следующим образом:

Команда:Всегоигр Побед Ничьих Поражений Всегоочков

Конкретный пример ввода-вывода приведён ниже.

Порядок вывода команд произвольный.

Пример входа:

3

Спартак;9;Зенит;10

Локомотив;12;Зенит;3

Спартак;8;Локомотив;15

Выход будет:

Спартак:2 0 0 2 0  
Зенит:2 1 0 1 3  
Локомотив:2 2 0 0 6

## Урок 7. Функции высшего порядка

**Задача 34:** Винни-Пух попросил Вас посмотреть, есть ли в его стихах ритм. Поскольку разобраться в его кричалках не настолько просто, насколько легко он их придумывает, Вам стоит написать программу. Винни-Пух считает, что ритм есть, если число слогов (т.е. число гласных букв) в каждой фразе стихотворения одинаковое. Фраза может состоять из одного слова, если во фразе несколько слов, то они разделяются дефисами. Фразы отделяются друг от друга пробелами. Стихотворение Винни-Пух вбивает в программу с клавиатуры. В ответе напишите “Парам пам-пам”, если с ритмом все в порядке и “Пам парам”, если с ритмом все не в порядке  
\*Пример:\*

```
**Ввод:** пара-ра-рам рам-пам-папам па-ра-па-да  
      **Вывод:** Парам пам-пам
```

**Задача 36:** Напишите функцию вывода таблицы умножения `print_operation_table(operation, num_rows=6, num_columns=6)`, которая принимает в качестве аргумента функцию, вычисляющую элемент по номеру строки и столбца. Аргументы `num_rows` и `num_columns` указывают число строк и столбцов таблицы, которые должны быть распечатаны. Нумерация строк и столбцов идет с единицы (подумайте, почему не с нуля). **Примечание:** бинарной операцией называется любая операция, у которой ровно два аргумента, как, например, у операции умножения.  
\*Пример:\*

```
**Ввод:** `print_operation_table(lambda x, y: x * y) `  
      **Вывод:**  
1 2 3 4 5 6
```

```
2 4 6 8 10 12  
3 6 9 12 15 18  
4 8 12 16 20 24  
5 10 15 20 25 30  
6 12 18 24 30 36
```

**Задание 1 необязательное** Сделайте локальный чат-бот с внешним хранилищем. Тема чат-бота - любая вам интересная.

**Задание 2 необязательное** Сделайте графический интерфейс для всех трех задач выше. Самый простой вариант - библиотека EasyGUI, потом посложнее это PyQt и Tkinter

**Задание 3 тяжелое необязательное** Сделайте приложение Калькулятор аналогично встроенному в Windows. Если будут силы то сделайте также переключение на режим инженерного расширенного режима.

## Урок 8. Работа с файлами

на Отлично в одного человека надо сделать консольное приложение Телефонный справочник с внешним хранилищем информации, и чтоб был реализован основной функционал - просмотр, сохранение, импорт, поиск, удаление.

Задача 38: Дополнить телефонный справочник возможностью изменения и удаления данных. Пользователь также может ввести имя или фамилию, и Вы должны реализовать функционал для изменения и удаления данных для отлично в группах надо выполнить или ТГ бот или ГУИ (это когда кнопки и поля ввода как в Виндовс приложениях) или БД  
ГУИ можно сделать просто на EasyGUI или Tkinter

## Урок 9. Работа с табличными данными

**Задача 40:** Работать с файлом `california_housing_train.csv`, который находится в папке `sample_data`. Определить среднюю стоимость дома, где кол-во людей от 0 до 500 (`population`).

**Задача 42:** Узнать какая максимальная `households` в зоне минимального значения `population`.

## Урок 10. Построение графиков

Делаем или это задание или  
доделываем ТГ Бот или доделываем  
Информационную систему или  
Калькулятор в двух режимах

| Задание 44 |

| --- |

| В ячейке ниже представлен код генерирующий DataFrame, которая состоит всего из 1 столбца. Ваша задача перевести его в `one hot` вид. Сможете ли вы это сделать без `get_dummies`?

```
import random
lst = ['robot'] * 10
lst += ['human'] * 10
random.shuffle(lst)
data = pd.DataFrame({'whoAmI':lst})
data.head()
```