

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГООБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Колледж экономики, управления и права

**Методические указания**

**по организации практических занятий и самостоятельных работ**

Программирование на Python

**Специальность**

*09.02.07 Информационные системы и программирование (по отраслям)*

Ростов-на-Дону

2023

Методические указания по Устройство и функционирование информационных систем разработаны с учетом ФГОС среднего профессионального образования специальности 09.02.04 Информационные системы (по отраслям), предназначены для студентов и преподавателей колледжа.

Методические указания определяют этапы выполнения работы на практическом занятии, содержат рекомендации по выполнению индивидуальных заданий и образцы решения задач, а также список рекомендуемой литературы.

Составитель (автор): преподаватель колледжа ЭУП С.П. Мантуленко

Рассмотрены на заседании предметной (цикловой) комиссии специальности 09.02.07 Информационные системы и программирование (по отраслям)

Протокол № \_\_\_ от «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2023 г

Председатель П(Ц)К специальности \_\_\_\_\_\_\_\_\_\_\_ С.В. Шинакова

личная подпись

и одобрены решением учебно-методического совета колледжа.

Протокол № \_\_\_ от «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2023 г

Председатель учебно-методического совета колледжа

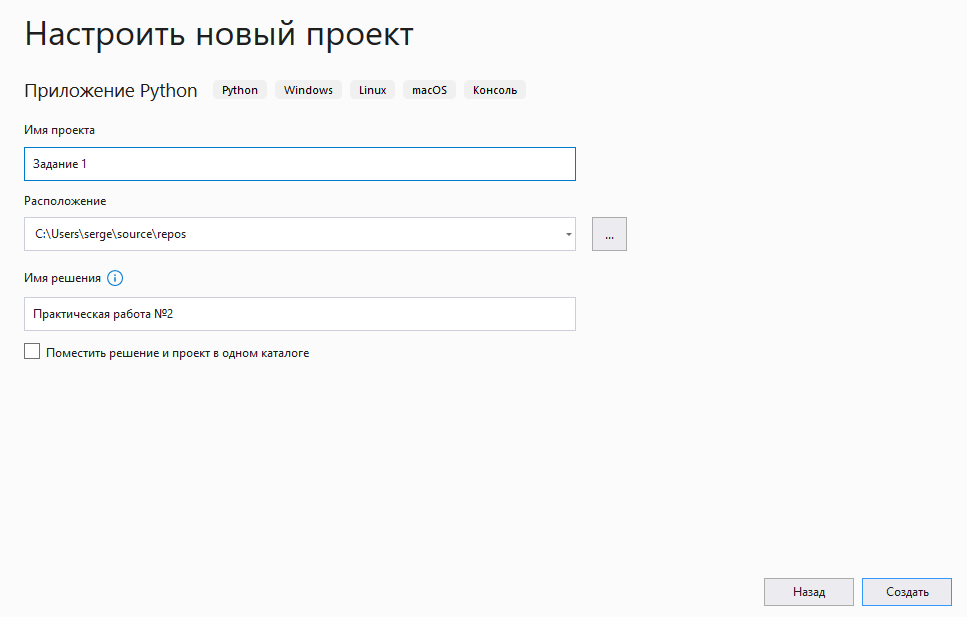
\_\_\_\_\_\_\_\_\_\_\_ С.В. Шинакова

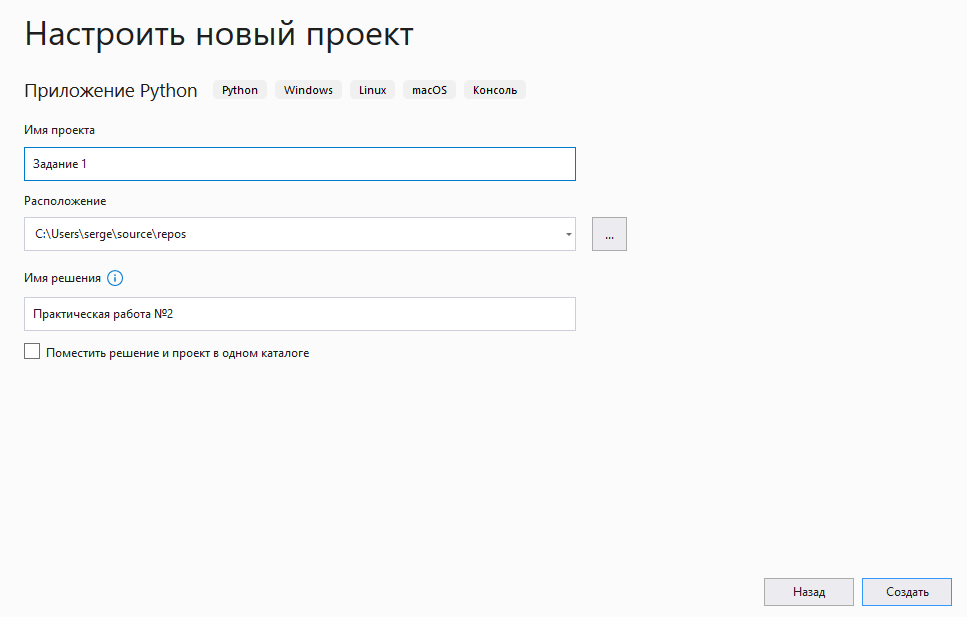
личная подпись

**Практическая работа №1-2**

**Решение элементарных математических задач**

Задание 1: запустите Microsoft Visual Studio и создайте новое решение, которое назовите *Практическая работа №2,* имя проекта *Задание 1*.



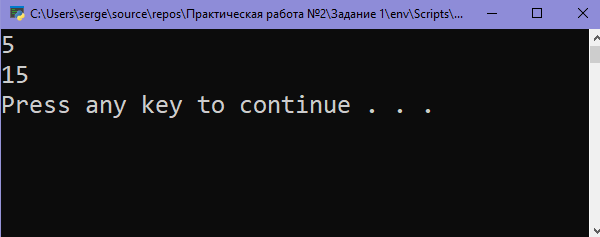
**

**Решим задачу:** найти значение функции y=2x+5 для любого значения Х.

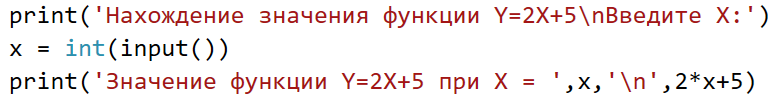
Задачи подобного рода на языке Python решаются буквально в пару строчек кода:



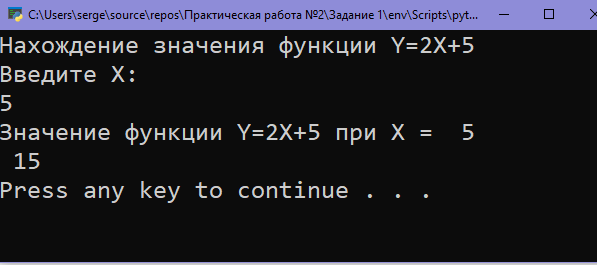
Но как будет выглядеть наша программа при выполнении?



Как видно, пользователь не догадывается, что ему нужно ввести и что он получил в ответ, поэтому давайте добавим несколько строчек пояснительного текста:



В результате мы получим:



*Вопрос: почему при вводе с клавиатуры значения Х мы указали «int(input())»?*

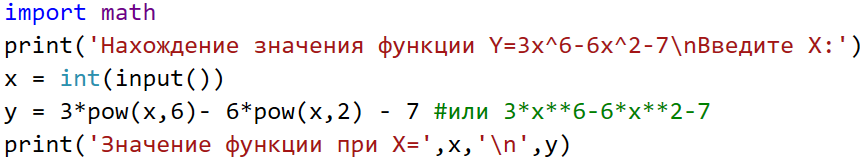
Задание 2: Найти значение функции y = при данном значении x. Добавьте новый проект в решение и назовите его *Задание 2*.

В данной задаче у нас появляются вычисления со степенями, *при чём больше, чем 2.* В языке python такую задачу можно решить несколькими способами.

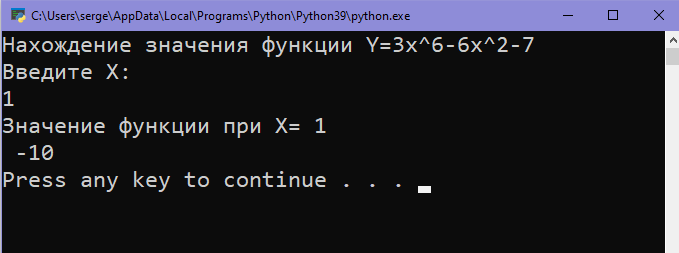
1 – Возведение в степень оператором «\*\*»: х\*\*2 (X2)

2 – Использование математической функции Pow: Pow(x,2).

Напишем следующий код:



Для просты проверки вычислений введем 1:

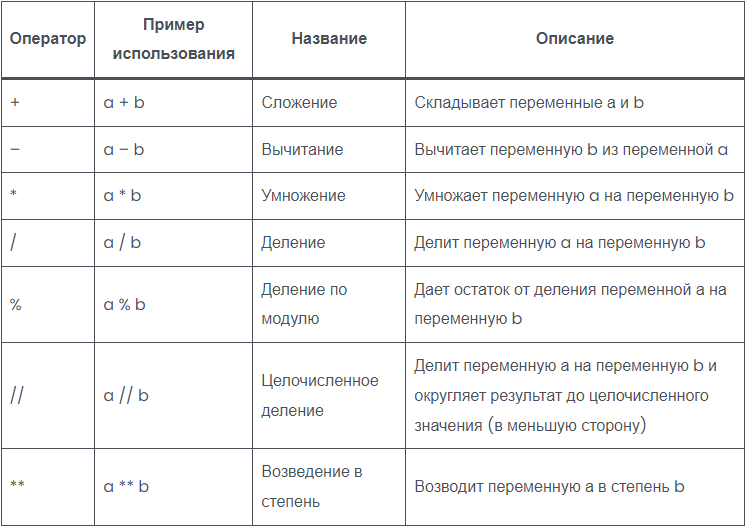


*Попробуйте воспользоваться первым способом возведения в степень. Изменился ли результат?*

Типы данных:

|  |  |  |  |
| --- | --- | --- | --- |
| **Тип данных** | **Значение** | **Определение в Python** | **Вариант использования** |
| Целые числа | -3, -2, -1, 0, 1, 2, 3 | int | a = int(input()) |
| Вещественные числа | -1.5, -1.1, 0.6, 1.7 | float | a = float(input()) |
| Комплексные числа | −5i, 3+2i | complex | a = complex(input()) |
| Булевы значения | True, False | True, False | flag = True |
| NoneType | None | None | a = None |
| Строка | 'abracadabra' | str | a = str(5) |
| Список | [1, 2, 3], ['a', 'b', 'c'] | list | a = list(('a', 'b', 'c')) |
| Кортеж | ('red', 'blue', 'green') | tuple | a = tuple(('red', 'blue', 'green')) |
| Изменяемое множество | {'black', 'blue', 'white'}, {1, 3, 9, 7} | set | a = set(('black', 'blue', 'white')) |
| Неизменяемое множество | {'red', 'blue', 'green'}, {2, 3, 9, 5} | frozenset | a = frozenset((2, 5, 3, 9)) |
| Диапазон | 0, 1, 2, 3, 4, 5 | range | a = range(6) |
| Словарь | {'color': 'red', 'model': 'VC6', 'dimensions': '30x50'} | dict | a = dict(color='red', model='VC6', dimensions='30x50') |

Математические операторы:



**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

1. Вычислить значения функции y = 4(x-3)6-7(x-3)3+2 при любом значении x.
2. Дан радиус окружности. Найти ее диаметр.
3. Даны два целых числа. Найти их среднее геометрическое. (Средним геометрическим называется квадратный корень из произведения этих чисел).
4. Даны катеты прямоугольного треугольника. Найти его гипотенузу.
5. Напишите программу, в которую вводится целое число, после чего на экран выводится следующее и предыдущее целое число. Например, при вводе числа 15 на экран должно быть выведено:

*Следующее за числом 15 число – 16.*

*Для числа 15 предыдущее число – 14.*

1. Дана сторона квадрата. Найти его периметр.
2. Даны два целых числа. Найти их среднее арифметическое;
3. Даны катеты прямоугольного треугольника. Найти его периметр.
4. Напишите программу, в которую вводится имя человека и выводится на экран приветствие в виде слова «Привет», после которого должна стоять запятая, введенное имя и восклицательный знак. После запятой должен стоять пробел, а перед восклицательным знаком пробела быть не должно.

**Практическая работа №3**

**Решение задач с использованием условного оператора**

Создайте новое решение *Практическая работа №3* и проект *Задание 1*.

Задание 1: Сравнить между собой 2 числа.

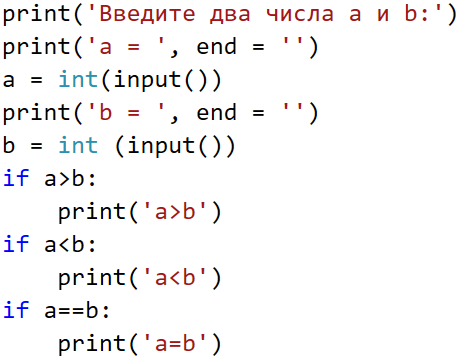
Мы помним, что 2 числа относительно друг друга могут иметь 3 логических состояния:

Первое больше второго;

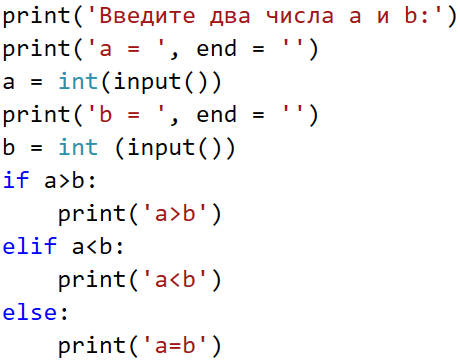
Второе больше первого;

Они равны.

Исходя из этого напишем следующий код:



В чем недостаток этого кода? Если первое условие окажется верным, то второе и третье будут проверяться в любом случае, а нам это не нужно, поэтому заменим второе условие на альтернативное условие:

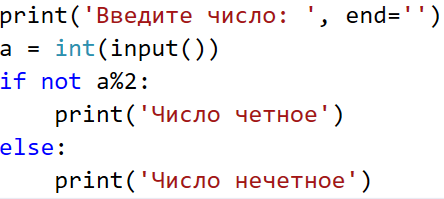


Запустите программу и проверьте результат ее выполнения.

Задание 2: Напишите программу, которая определяет четность числа.

Создадим новый проект и добавим его в наше решение.

Напишем следующий код:



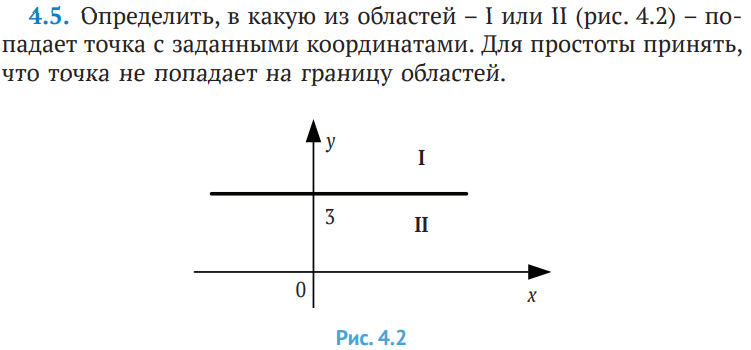
Запустите программу и проверьте результат её выполнения.

*Вопрос: почему в качестве условия использовалось данное выражение: «not a%2»?*

**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

1. Дано натуральное число. Определить оканчивается ли оно цифрой 7
2. Дано двузначное число. Определить: а)  является ли сумма его цифр двузначным числом; б)  больше ли числа «а» сумма его цифр.
3. Определить, является ли треугольник со сторонами a, b, c: а)  равносторонним; б)  равнобедренным
4. Даны три целых числа. Вывести на экран те из них, которые являются четными.
5. Даны три вещественных числа. Возвести в квадрат те из них, значения которых неотрицательны.
6. Даны четыре вещественных числа. Определить, сколько из них отрицательных. Оператор цикла не использовать.
7. Дано двузначное число. Определить: а)  какая из его цифр больше: первая или вторая; б)  одинаковы ли его цифры.
8. Дано двузначное число. Определить: а)  кратна ли трем сумма его цифр; б)  кратна ли сумма его цифр числу «а».
9. Даны два числа. Если квадратный корень из второго числа меньше первого числа, то увеличить второе число в пять раз.
10. Даны три вещественных числа. Вывести на экран те из них, которые принадлежат интервалу [1.6, 3.8].
11. Даны целые положительные числа a, b, c. Если существует треугольник со сторонами a, b, c, то определить, является ли он прямоугольным.



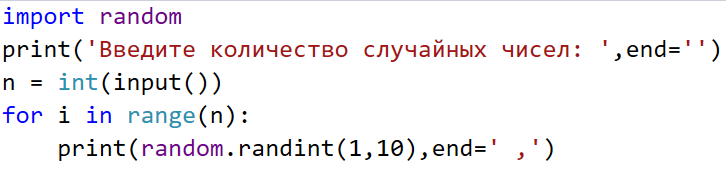
12.

**Практическая работа №4-5**

**Решение задач с использованием циклов**

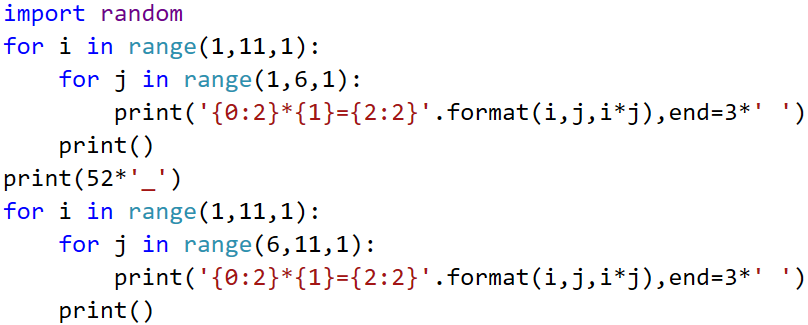
Создайте новое решение и проект.

Задание 1: напишите программу, которая выведет n-е количество случайных чисел. Количество чисел вводит пользователь.



Чтобы иметь возможность генерировать случайные числа, мы импортировали библиотеку random. Метод randint генерирует случайное число из диапазона, указанного в скобках.

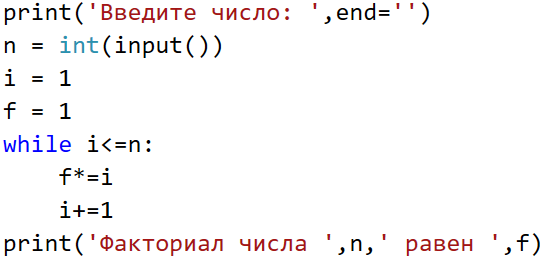
Задание 2: вывести на экран таблицу умножения по 5 столбиков.



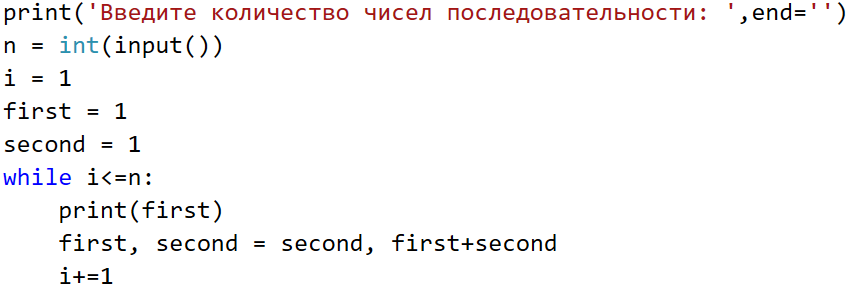
*Вопрос: что означает строчка «print(52\*’\_’)»?*

Задание 3: напишите программу нахождения факториала с помощью цикла while.

Конструкция условного цикла такая же, как и во многих других языках программирования. Код будет выглядеть следующим образом:

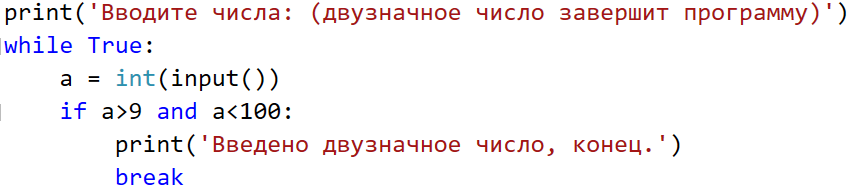


Задание 4: Найдите n чисел Фибоначчи, n вводится пользователем.

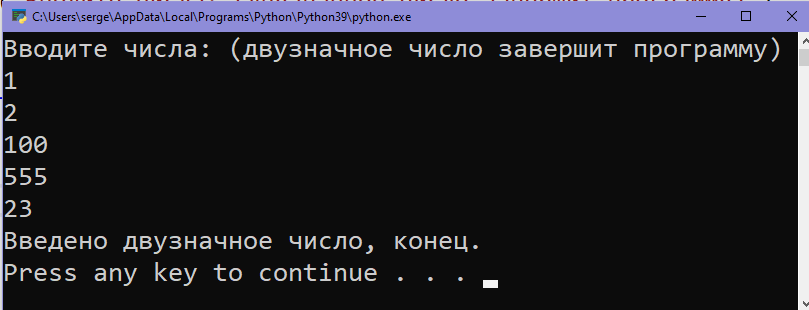


*Вопрос: что означает строчка «first, second = second, first+second»?*

Задание 5: написать программу непрерывного ввода чисел. Если пользователь ввел двузначное число, программа останавливается.



Результат:



**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

**Часть 1**

1. Найти

а)  произведение всех целых чисел от 7 до 14;

б)  произведение всех целых чисел от a до 15 (значение a вводится с клавиатуры; 1 ≤ a ≤ 15);

1. Последовательность Фибоначчи образуется так: первый и второй члены последовательности равны 1, каждый следующий равен сумме двух предыдущих (1, 1, 2, 3, 5, 8, 13, ...). Дано натуральное число n (n ≥ 3). Найти n-й член последовательности Фибоначчи.
2. Вывести на экран все нечетные двузначные числа, у которых последняя цифра равна 3 или 7.
3. Дано натуральное число. Выяснить, является ли оно простым (простым называется натуральное число, большее 1, не имеющее других делителей, кроме единицы и самого себя).
4. Написать программу возведения числа «а» в степень «b» без использования функции возведения в степень.
5. Дано целое число A и N (N>0). Найти значение выражения: А+А2-А3+…-АN

**Часть 2**

1. Даны натуральные числа a и b (a > b). Определить результат целочисленного деления a на b, не используя стандартную операцию целочисленного деления.
2. Напечатать те натуральные числа, квадрат которых не превышает заданное число n
3. Разработать фрагмент программы, в котором пользователь должен ввести четное число. В случае ввода нечетного числа на экран должно выводиться сообщение об ошибке, после чего действия должны повторяться до ввода правильного значения.
4. Подготовьте фрагмент программы, в котором должны вводиться 10 чисел. Если будет введено число 0, ввод должен прекратиться.
5. Дано натуральное число. Определить:

а)  количество цифр 3 в нем;

б)  сколько раз в нем встречается последняя цифра;

в)  количество четных цифр в нем.

1. Напечатать все нечетные числа из интервала [10, 100]. Оператор цикла с параметром не использовать.
2. Подготовить фрагмент программы, в которой пользователь должен ввести установленный пароль в виде целого числа. В случае ввода неправильного пароля на экран должно выводиться сообщение об ошибке, после чего действия должны повторяться до ввода правильного значения. После этого на экран должно выводиться некоторое приветствие.
3. Напечатать числа 1.0, 1.5, 2.0, …, 13.5. Оператор цикла с параметром не использовать.
4. Дано натуральное число. Определить:

а)  сумму его цифр, больших пяти;

б)  произведение его цифр, больших семи;

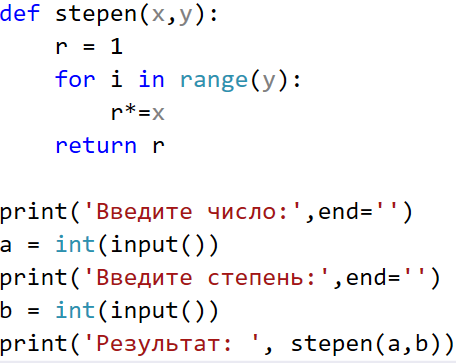
в)  сколько раз в нем встречаются цифры 0 и 5 (всего)

**Практическая работа №6-7**

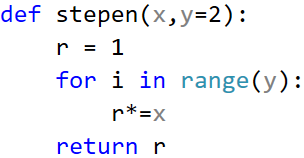
**Решение задач с использованием пользовательских функций**

Задание 1: Напишите функцию, возведения числа в степень (представим, что такой функции не существует)

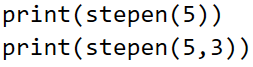
Как известно, показатель степени указывает на то, сколько раз число было умножено само на себя. Для функции нам понадобится два параметра, возводимое число и степень. Так как возводимое число будет умножаться определенное количество раз на само себя (то есть будет выполняться одно и то же действие) и это количество нам известно, то мы можем использовать цикл for:



Давайте изменим функцию таким образом: если мы не укажем параметр степени, то число по умолчанию будет возведено в квадрат. Для этого мы используем необязательный параметр:

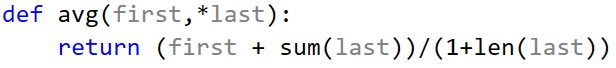


Вызовем функцию с разным количеством параметров:



В результате на экран выведется 25 и 125.

Задание 2: Напишите функцию, которая будет принимать любое количество параметров и возвращать их среднее арифметическое. Чтобы функция могла использовать переменное количество параметров, необходимо использовать аргумент «\*»:



*Вопрос: почему сумму элементов делят на (1+len(last))?*

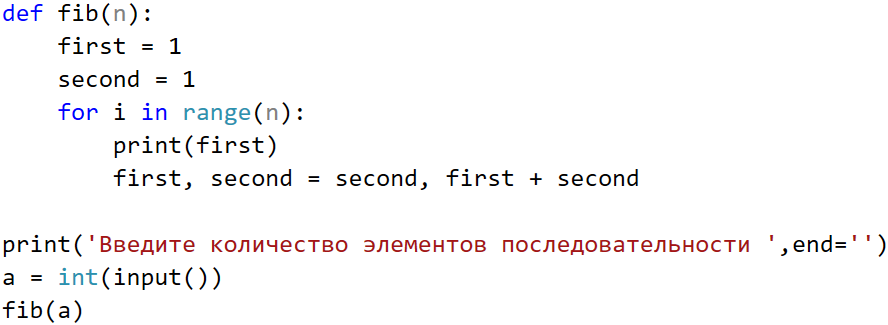


На экран выведется 3 и 2.

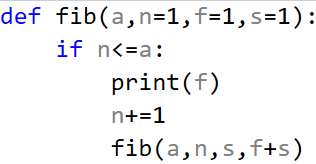
Задание 3: Напишите функцию, которая выведет n чисел последовательности Фибоначчи.

Функции могут не возвращать никакого значения, тогда в них не нужно указывать слово return.

Такая задачу мы решали в одной из предыдущих практик, оформим код согласно условию задачи:

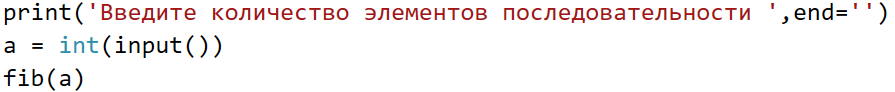


Давайте перепишем эту функцию с использованием рекурсии (функция вызывает саму себя).

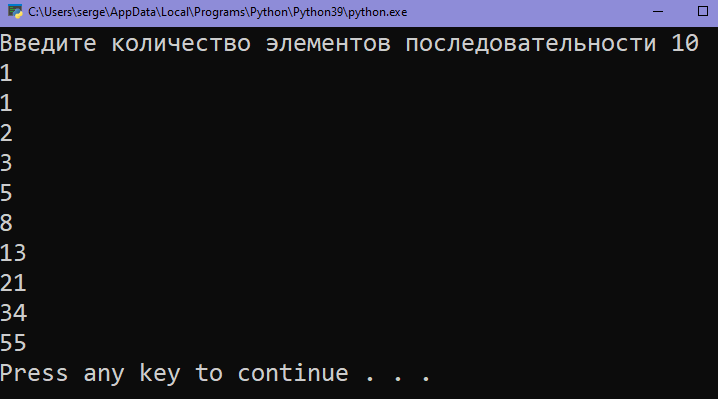


Как мы видим, кроме обязательного параметра, функцию пополнили необязательные параметры: n – номер ткущей итерации, f и s – первое и второе число последовательности соответственно. С помощью условного оператора проверяется условие следующего вызова функции (если не указать условие, функция будет вызвать саму себя до ошибки).

Впрочем, несмотря на увеличение количества параметров, функция вызывается с одним обязательным параметром «а»:



Результат не изменился:



**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

**Часть 1**

1. Составить функцию, которая определяет сколько цифр в введенном числе (скольки-значное число?).
2. Вычислить наименьшее значение Y(В[i])=5\*B[i]3+3\*B[i]2-B[i], если B[i] задано массивом *int[] B = new int[N]*.
3. Даны числа S, T. Получить с использованием функции пользователя с параметрами:

G(1, sin(S))+2G(T\*S, 24)-G(5, -S), где G(A,B) = (2A+B2)/(2A\*B+5B)

1. Составить функцию, принимающую в качестве параметров коэффициенты квадратного уравнения и находящую его корни.
2. Написать функцию, которая определяет, является ли число В делителем числа А.
3. Написать функцию, которая в качестве параметров принимает размеры трех сторон треугольника и определяет, является ли данный треугольник прямоугольным.
4. Составить функцию, рассчитывающую сумму цифр введенного числа.
5. Составить функцию, которая выводит все делители введенного числа.
6. Написать функцию, которая в качестве параметров принимает координаты точки на плоскости и в результате выводит сообщение, в какой четверти находится точка.

**Часть 2**

1. Создайте функцию, которая возвращает логический тип и проверяет, кратна ли сумма цифр двузначного числа числу А, введенному с клавиатуры. Проверить возвращаемое значение и вывести соответствующее сообщение.
2. Напишите функцию, которая ищет в числе количество единиц двоичной записи. В программу вводится десятичное число. Программа должна вывести количество единиц в двоичной записи.
3. Обязательная задача\* Маска – это число, которое в двоичной системе счисления записывается по следующему правилу: Сначала единицы, а затем нули. (Используется в компьютерных сетях) Например, число 192 является маской, так как в двоичном виде записывается как 11000000, а число 193 не является маской, так как в двоичной записи имеет вид 11000001, что не соответствует правилу, что в двоичной записи числа сначала должны идти единицы, а затем нули. С клавиатуры вводятся N целых чисел (Число N также вводится с клавиатуры). Напишите программу, которая среди них находит и выводит только те, которые являются маской. Для решения этой задачи, напишите логическую функцию Mask, которая принимает значение истина, если число является маской, и ложь в противном случае.
4. Напишите логическую функцию, которая возвращает истину, если в числе две последние цифры одинаковые. Например, функция вернет истину, если будет введено число 377 или 12855. В противном случае функция возвращает ложь. Программа получает на вход целое число. Программа должна вывести YES, если функция возвращает истину и NO в обратном случае.
5. Напишите функции, которые по входным двум целым числам a и b возвращают все значения формул сокращенного умножения (квадрат разности, квадрат суммы и разность квадратов)
6. Напишите функцию count(x,y), которая возвращает количество цифр «y» в целом числе «x». Переменные x и y — целого типа. Например, значение функции count(177,7) будет равно 2. Проверьте работу функции при входном целом числе. Программа должна работать как для положительных, так и для отрицательных чисел.

**Практическая работа №8**

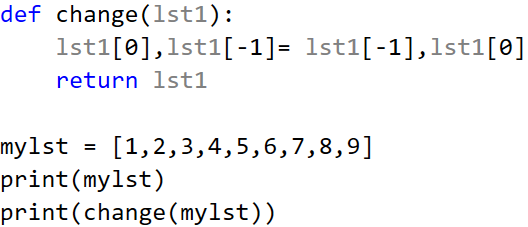
**Решение задач с использованием списков**

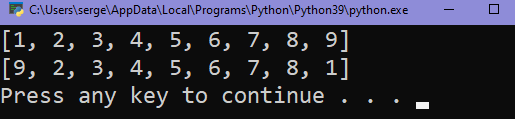
Операции над списками:

|  |  |
| --- | --- |
| **Метод** | **Что делает** |
| **list.append**(x) | Добавляет элемент в конец списка |
| **list.extend**(L) | Расширяет список list, добавляя в конец все элементы списка L |
| **list.insert**(i, x) | Вставляет на i-ый элемент значение x |
| **list.remove**(x) | Удаляет первый элемент в списке, имеющий значение x. ValueError, если такого элемента не существует |
| **list.pop**([i]) | Удаляет i-ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент |
| **list.index**(x, [start [, end]]) | Возвращает положение первого элемента со значением x (при этом поиск ведется от start до end) |
| **list.count**(x) | Возвращает количество элементов со значением x |
| **list.sort**([key=функция]) | Сортирует список на основе функции |
| **list.reverse**() | Разворачивает список |
| **list.copy**() | Поверхностная копия списка |
| **list.clear**() | Очищает список |

Задание1: Составьте функцию, которая принимает в качестве параметра список и меняет местами его первый и последний элементы.

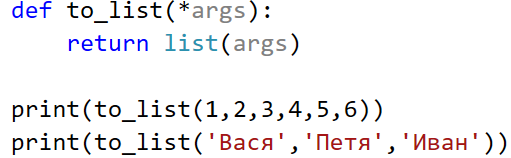
Так как списки являются изменяемыми типами данных, то мы можем использовать обычное присвоение:

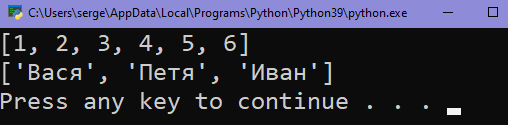




Задание 2: Функция to\_list() принимает неограниченное количество параметров. Обработайте их так, чтобы на выходе получить список из этих элементов.

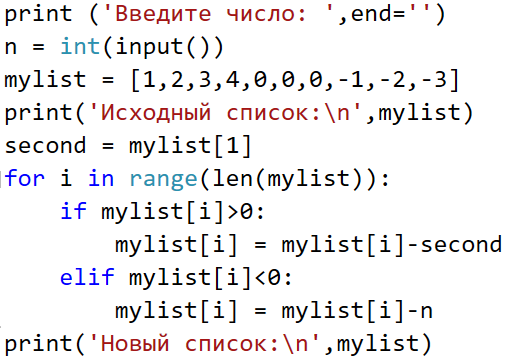
Давайте вспомним функции, которое могут принимать переменное количество параметров. Такие параметры выделяются аргументов «\*». То есть наша функция будет иметь следующий заголовок: «def to\_list(\*args)». А далее мы используем функцию list(), которая формирует список:



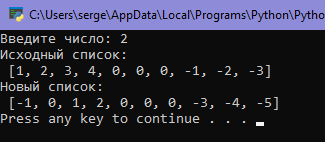


Как видно, при выполнении программы параметры, которые мы передали функции, вывелись в виде списка.

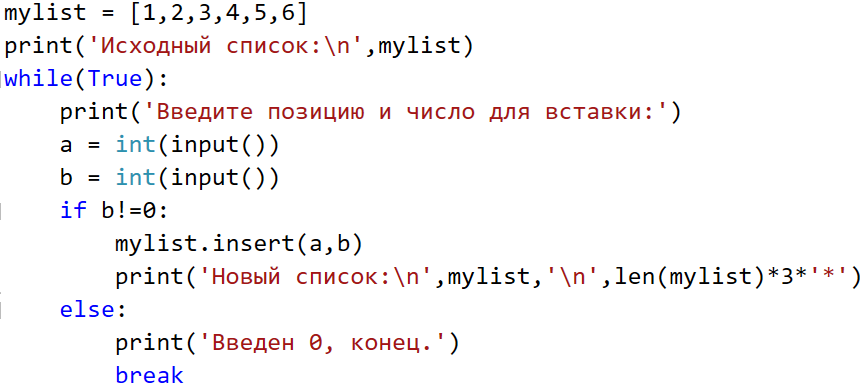
1. Задание 3: Дан массив вещественных чисел, из всех положительных элементов вычесть элемент с номером k[1], из всех отрицательных – число n. Нулевые элементы оставить без изменения.

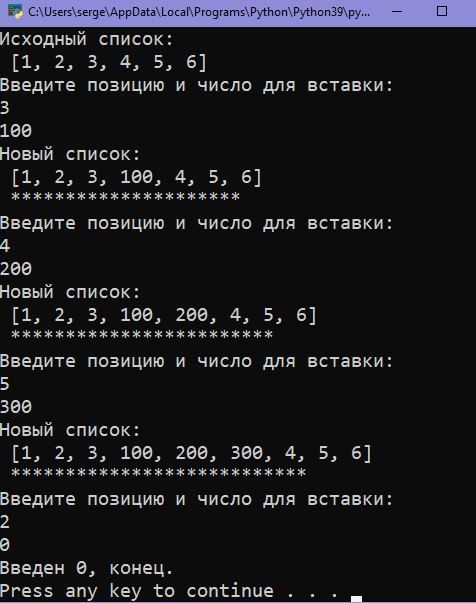


*Вопрос: зачем используется переменная «second»?*



Задание 4: Напишите программу, которая будет добавлять введенное пользователем число в введенную позицию в списке. Программа должна выполняться, пока не будет введен 0.





**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

1. Заполнить список десятью первыми членами арифметической прогрессии с известным первым членом прогрессии А и ее разностью Р;
2. Дан список целых чисел. Все элементы, оканчивающиеся цифрой 4, уменьшить вдвое.
3. Найти:

а) сумму всех элементов списка;

б) произведение всех элементов списка;

1. Список предназначен для хранения значений весов двадцати человек. С помощью датчика случайных чисел заполнить массив целыми значениями, лежащими в диапазоне от 50 до 100 включительно.
2. Вывести элементы списка на экран в обратном порядке.
3. Дан список вещественных чисел, ко всем нулевым элементам прибавить n, из всех положительных элементов вычесть А, ко всем отрицательным прибавить В.
4. Дан список целых чисел. Все четные элементы заменить на их квадраты, а нечетные удвоить.
5. Найти:

а) сумму квадратов всех элементов списка;

б) среднее арифметическое всех элементов массива;

1. Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер первого числа в наборе, большего K. Если таких чисел нет, то вывести 0.
2. Задан список из десяти городов. Подсчитать количество названий, которые оканчиваются буквой В.

**Практическая работа №9**

**Решение задач с использованием строк**

|  |  |
| --- | --- |
| **Функция или метод** | **Назначение** |
| **S1 + S2** | Конкатенация (сложение строк) |
| **S1 \* 3** | Повторение строки |
| **S[i]** | Обращение по индексу |
| **S[i:j:step]** | Извлечение среза |
| **len**(S) | Длина строки |
| **S.index**(str, [start],[end]) | Поиск подстроки в строке. Возвращает номер первого вхождения или вызывает ValueError |
| **S.rindex**(str, [start],[end]) | Поиск подстроки в строке. Возвращает номер последнего вхождения или вызывает ValueError |
| **S.replace**(шаблон, замена[, maxcount]) | Замена шаблона на замену. maxcount ограничивает количество замен |
| **S.split**(символ) | Разбиение строки по разделителю |
| **S.isdigit**() | Состоит ли строка из цифр |
| **S.isalpha**() | Состоит ли строка из букв |
| **S.isalnum**() | Состоит ли строка из цифр или букв |
| **S.islower**() | Состоит ли строка из символов в нижнем регистре |
| **S.isupper**() | Состоит ли строка из символов в верхнем регистре |
| **S.isspace**() | Состоит ли строка из неотображаемых символов (пробел, символ перевода страницы ('\f'), "новая строка" ('\n'), "перевод каретки" ('\r'), "горизонтальная табуляция" ('\t') и "вертикальная табуляция" ('\v')) |
| **S.istitle**() | Начинаются ли слова в строке с заглавной буквы |
| **S.upper**() | Преобразование строки к верхнему регистру |
| **S.lower**() | Преобразование строки к нижнему регистру |
| **S.startswith**(str) | Начинается ли строка S с шаблона str |
| **S.endswith**(str) | Заканчивается ли строка S шаблоном str |
| **S.join**(список) | Сборка строки из списка с разделителем S |
| **ord**(символ) | Символ в его код ASCII |
| **chr**(число) | Код ASCII в символ |
| **S.capitalize**() | Переводит первый символ строки в верхний регистр, а все остальные в нижний |
| **S.lstrip**([chars]) | Удаление пробельных символов в начале строки |
| **S.rstrip**([chars]) | Удаление пробельных символов в конце строки |
| **S.strip**([chars]) | Удаление пробельных символов в начале и в конце строки |
| **S.swapcase**() | Переводит символы нижнего регистра в верхний, а верхнего – в нижний |
| **S.title**() | Первую букву каждого слова переводит в верхний регистр, а все остальные в нижний |
| **S.zfill**(width) | Делает длину строки не меньшей width, по необходимости заполняя первые символы нулями |
| **S.ljust**(width, fillchar=" ") | Делает длину строки не меньшей width, по необходимости заполняя последние символы символом fillchar |
| **S.rjust**(width, fillchar=" ") | Делает длину строки не меньшей width, по необходимости заполняя первые символы символом fillchar |

**Самостоятельная работа**

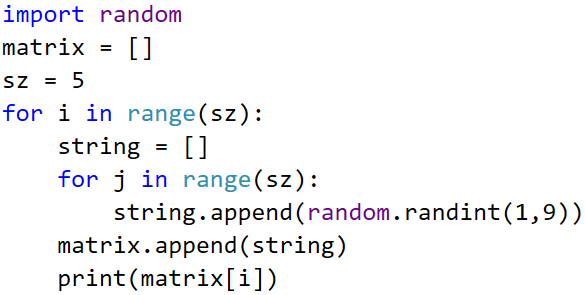
1. Вставить в предложение с 3-го символа слово «кит».
2. Удалить из предложения 3 символа, начиная с 6-го.
3. Скопировать из предложения 3 символа, начиная со 2-го.
4. Подсчитать сколько раз встречается буква «м», предлог «не» в предложении.
5. Выяснить, есть ли в предложении хотя бы одна пара одинаковых символов.
6. Удалить из предложения все пробелы.
7. Вставить в предложение пробелы после каждой буквы «а».
8. Проверить, есть ли в предложении запятые.
9. Определить, сколько фамилий в списке начинаются с буквы «А».
10. Определить, сколько фамилий в списке имеют окончание «ов».
11. Определить, сколько фамилий в списке заканчиваются на букву «н».
12. Определить, сколько букв в самой длинной фамилии списка.
13. Определить, сколько букв в самой короткой фамилии списка.
14. Найти самую длинную фамилию в списке. Если таких фамилий несколько, то распечатать их в одну строку.
15. Найти в списке все фамилии, начинающиеся с букв «В» или «Г».
16. Найти в списке все фамилии, имеющие окончание «ев».
17. Найти в списке все фамилии, начинающиеся со слога «Ма».
18. Определить, сколько фамилий в списке состоят менее чем из 6 букв.
19. Определить, сколько фамилий в списке состоят из 8 или 9 букв.
20. Выяснить, имеется ли в списке фамилия «Ганеев». Если имеется, то исправить ее на «Ганиев».
21. Выяснить, сколько раз в списке встречается фамилия «Иванов».
22. Поменять местами первую и последнюю фамилии в списке.
23. Заменить вторую фамилию в списке на «Нигматуллин» и удалить из списка третью фамилию.
24. Выяснить, имеются ли в списке фамилии «Петров». Если имеются, то удалить их из списка.
25. Упорядочить список в алфавитном порядке.
26. Определить, имеются ли в списке однофамильцы.
27. Определить, четна ли сумма цифр, занимающих во введенном числе нечетные позиции.
28. Даны два слова одинаковой длины. Присвоить переменной k число, равное количеству попарно одинаковых букв.
29. Даны два слова. Сколько раз во втором слове встречается первая буква первого слога.
30. Задан список из десяти городов. Поменять местами названия двух городов, названия которых оканчиваются сочетанием букв «град».
31. Имеется некоторая последовательность символов. Образовать новую последовательность, включив в нее символы исходной, кроме символов «ы» и «э».
32. Задан список из десяти городов. Подсчитать количество названий, в которых есть по две буквы «а».
33. Задан список из десяти городов. Поменять местами названия любых двух городов, заканчивающихся буквой «а».
34. Даны два слова разной длины. Присвоить переменной m число, равное количеству попарно различных букв.
35. Имеется некоторый текст. Образовать из него новый, в который включить информацию, заключенную между пробелом и запятой.
36. Задан список из десяти городов. Присвоить переменной t название последнего из городов, которое содержит более 4-х букв.
37. Задан список из 5 имен девочек. Присвоить переменной d имя с наименьшим числом букв.
38. Задан список из десяти городов. Присвоить переменной s название города с максимальным числом букв.
39. Задан текст из 20 символов латинского алфавита. Подсчитать в нем количество гласных букв.
40. Задан список из десяти городов. Поменять местами названия первого города и любого другого, которое содержит более семи букв.
41. Из двух восьмибуквенных слов образовать последовательность букв, в которой должны чередоваться буквы первого и второго слова.
42. Задан список из десяти городов. Поменять местами названия последнего города и любого из городов, название которого оканчивается на букву «к».
43. Имеется некоторая последовательность символов. Образовать новую последовательность, включив в нее символы исходной в обратном порядке.
44. Задан список из десяти городов. Подсчитать количество названий, в которых есть буква «Д».
45. Задан список из десяти городов. Поменять местами названия самого длинного и самого короткого слова.
46. Образовать последовательность символов, включив в нее символы данной последовательности, расположенные на нечетных позициях.
47. Задан текст из символов латинского алфавита, содержащий букву а. Написать все символы, расположенные за первой буквой «а» до ее второго вхождения или до конца текста.
48. Вводится строка – фамилия, имя и отчество учащегося. Вывести на экран преобразованную строку: оставить только фамилию и инициалы.
49. Задан список из десяти городов. Присвоить переменной t название последнего из городов, которое содержит более четырех букв.
50. Задан список из 10 городов. Поменять местами названия первого и последнего города.
51. Определить наибольшую цифру введенного натурального числа.
52. Определить количество символов введенной строки, ASCII-коды которых больше 70.
53. Изменить введенную строку цифр, разместив сначала цифры занимающие нечетные места, потом четные.

**Практическая работа №10**

**Решение задач с использованием матриц**

Так как в python, по сути, нет конструкции «массив», то соответственно нет структуры «двумерный массив». Вместо этого мы будем работать со вложенными списками.

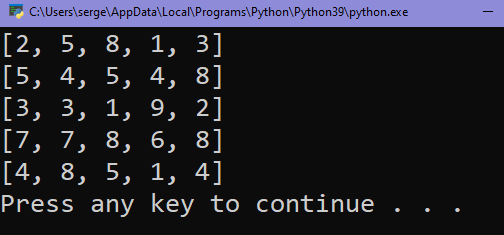
Задание 1: Составьте программу, которая будет автоматически создавать и выводить матрицу из случайных чисел.



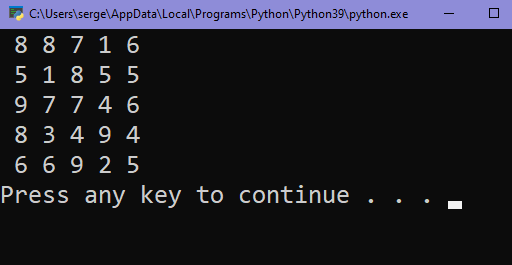
Давайте разберем все по порядку:

1. Для работы с генератором случайных чисел мы импортировали библиотек random. Затем создали пустой список matrix и задали переменную размера матрицы sz;
2. На каждой итерации внешнего цикла у нас будет создаваться новая пустая строк и после выполнения внутреннего цикла, она будет добавляться в матрицу;
3. Внутренний цикл формирует каждую новую строку из случайных чисел.

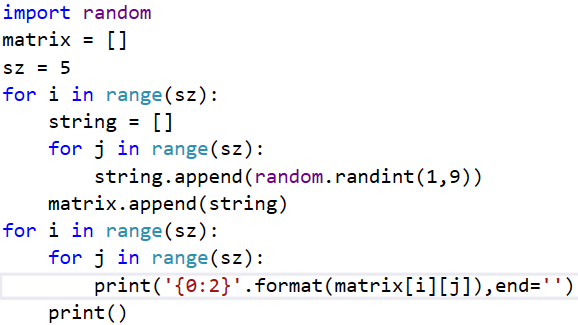
В результате мы получим следующую матрицу:



Как видно, у нас вывелся набор списков. Давайте изменим вывод следующим образом:



Для этого изменим наш код:

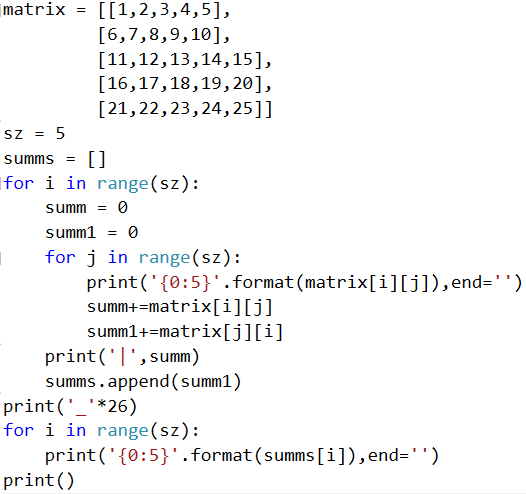


Задание 2: Напишите программу, которая будет рассчитывать сумму элементов каждой строки и столбца матрицы.

Заполним матрицу статично:

matrix = [[1,2,3,4,5],[6,7,8,9,10],[11,12,13,14,15],[16,17,18,19,20],[21,22,23,24,25]]

Как видно, каждый элемент списка «matrix» представляет собой еще один список. Для решения задачи напишем следующий код:



**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

1. Найти максимальный и минимальный элемент матрицы
2. Найти сумму элементов главной и побочной диагоналей квадратной матрицы.
3. Подсчитать количество строк, содержащих отрицательные элементы.
4. Найти максимальные элементы во всех столбцах матрицы.
5. Заполнить квадратную матрицы значениями, введенными с клавиатуры. Найти номер первой строки, содержащей максимальный элемент в матрице.
6. Заполнить матрицу элементами от -10 до +10. Найти сумму модулей элементов для каждого столбца матрицы.
7. Подсчитать количество столбцов, содержащих отрицательные элементы.
8. Найти минимальные элементы во всех строках матрицы.
9. \*Характеристикой столбца целочисленной матрицы назовем сумму модулей его отрицательных нечетных элементов. Переставляя столбцы заданной матрицы, расположить их в соответствии с ростом характеристик (оформить в виде функции)
10. \*Дана целочисленная квадратная матрица. Определить: максимум среди сумм элементов диагоналей, параллельных главной диагона­ли матрицы (оформить в виде функции).
11. \*Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик (оформить в виде процедуры).

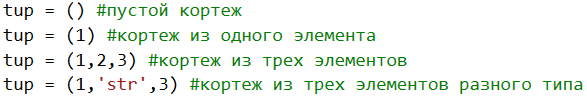
**Практическая работа №11**

**Решение задач с использованием кортежей и словарей**

Кортежи – это еще один из типов последовательностей. Но в отличие от строк, которые состоят только из символов, кортеж могут содержать элементы любой природы. В кортеже вы можете хранить, например, фамилии сотрудников, марки автомобилей, номера телефонов и т.д.

В отличие от списка, кортежи относятся к неизменяемым типам данных. Это означает, что вы можете получить элемент по индексу, но не можете его изменить.

**Создание кортежей:**



Кроме того, можно использовать функцию tuple():

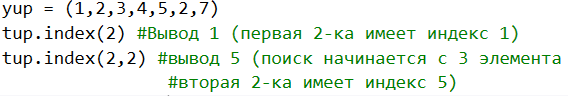


Обратиться к элементу кортежа можно по его индексу (начиная с 0).

Методы кортежей:

index(<значение>[,<начало>,<конец>])

count(<значение>)



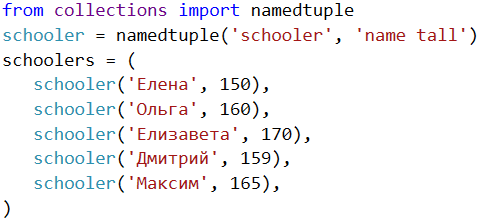
Второй метод подсчитывает количество элементов в кортеже с указанным значением:



Кортеж можно использовать как условие. Пустой кортеж интерпретируется как false, а кортеж, содержащий хотя бы один элемент, как true.

Задание 1: Кортеж состоит из списка учащихся, где каждый учащийся является кортежем с информацией об имени и росте ученика. Определить средний рост в классе и вывести список учеников, чей рост выше среднего.

Для решения этой задачи создадим именованный кортеж. Для этого импортируем из библиотеки collections класс namedtuple:

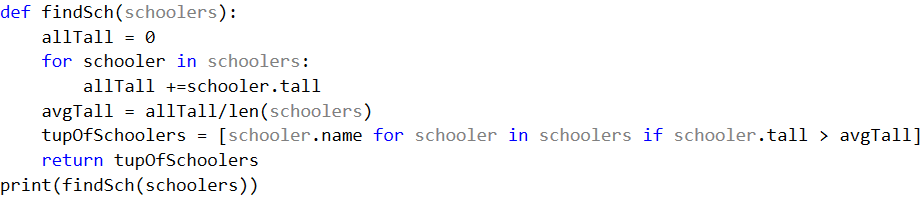


Функция namedtuple принимает два параметра:

1 – имя кортежа;

2 –список полей кортежа.

Теперь создадим функцию для расчета среднего роста и на его основе нахождения ребят, выше среднего роста, которая в качестве параметра принимает наш кортеж со школьниками:



Обратите внимание, что к полям именованного кортежа можно обращаться через точку.

Запустите программу и проверьте результат.

**Методы работы со словарями:**

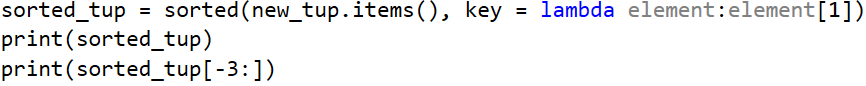
|  |
| --- |
| **dict.clear**() - очищает словарь. |
| **dict.copy**() - возвращает копию словаря. |
| **dict.fromkeys**(seq[, value]) - создает словарь с ключами из seq и значением value (по умолчанию None). |
| **dict.get**(key[, default]) - возвращает значение ключа, но если его нет, возвращает default (по умолчанию None). |
| **dict.items**() - возвращает пары (ключ, значение). |
| **dict.keys**() - возвращает ключи в словаре. |
| **dict.pop**(key[, default]) - удаляет ключ и возвращает значение. Если ключа нет, возвращает default (по умолчанию бросает исключение). |
| **dict.popitem**() - удаляет и возвращает пару (ключ, значение). |
| **dict.setdefault**(key[, default]) - возвращает значение ключа, но если его нет, создает ключ со значением default (по умолчанию None). |
| **dict.update**([other]) - обновляет словарь, добавляя пары (ключ, значение) из other. |
| **dict.values**() - возвращает значения в словаре. |

Задание 2: Дана строка в виде случайной последовательности чисел от 0 до 9. Требуется создать словарь, который в качестве ключей будет принимать данные числа (т. е. ключи будут типом int), а в качестве значений – количество этих чисел в имеющейся последовательности. Для построения словаря создайте функцию, принимающую строку из цифр. Функция должна возвратить словарь из 3-х самых часто встречаемых чисел.

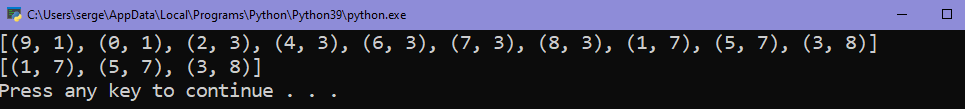
Создадим функцию, где первым делом сформируем словарь, где ключами будут выступать цифры из последовательности, а значениями – количество этих цифр в последовательности:



Затем отсортируем словарь по значениям в порядке возрастания. Для этого с помощью метода items() формируем пары «ключ, значение» в виде кортежей по всем элементам словаря. Т. к. нужно сортировать по значениям, берем второй элемент кортежей. (element:element[1]):



Выведем на экран отсортированный словарь и 3 самые встречающиеся цифры последовательности:



**Самостоятельная работа**

***Все задачи помещаются в одно решение!***

1. Напишите функцию tpl\_sort(), которая сортирует кортеж, состоящий из целых чисел, по возрастанию и возвращает его. Если хотя бы один элемент не является целым числом, то функция возвращает исходный кортеж.
2. Напишите функцию slicer(), которая на вход принимает кортеж и случайный элемент. Она должна вернуть новый кортеж, начинающийся с первого появления элемента в нем и заканчивающийся вторым его появлением включительно. Если элемента нет вовсе – вернуть пустой кортеж. Если элемент встречается только один раз, то вернуть кортеж, который начинается с него и идет до конца исходного.
3. Напишите функцию, которая на вход получает кортеж целых чисел. В результате выполнения этой функции будет получен кортеж уникальных элементов списка в обратном порядке.
4. Известно, что кортеж – это неизменяемый список. Попробуйте обойти это правило и составьте функцию, которая будет удалять первое появление определенного элемента из кортежа по значению и возвращать кортеж без него. Если в кортеже нет это элемента, то он должен вернуться в исходном виде.
5. Создайте кортеж из 7-ми именованных кортежей учащихся колледжа. В именованном кортеже будут присутствовать следующие поля: имя студента, возраст, оценка за семестр, город проживания. Функция good\_students() будет принимать этот кортеж, вычислять среднюю оценку по всем учащимся и выводить на печать следующее сообщение: "Ученики {список имен студентов через запятую} в этом семестре учется хорошо!". В список студентов, которые выводятся по результатам работы функции, попадут лишь те, у которых оценка за семестр равна или выше средней по всем учащимся.
6. Напишите функцию, которая принимает аргумент в виде списка и возвращает словарь, в котором каждый элемент списка является и ключом, и значением.
7. Создайте словарь, содержащий одну пару ключ-значение(«first-1»). Придумайте функцию, которая принимает переменное количество параметров ключ-значение (\*\*kwargs) и обновляет созданный словарь.

**Практическая работа №12**

**Построение графиков математических функций**

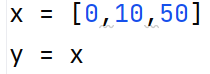
Задание 1: Построить график линейной функции y=x.

Для построения графиков нам необходимо импортировать библиотеку matplotlib:

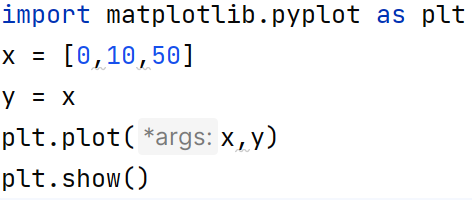


Название plt общепринятое для модуля pyplot, но вы можете использовать любое сокращение.

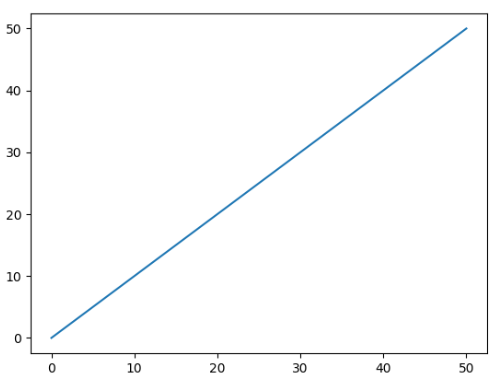
Зададим список значений аргумента функции, и так как, значения y равны x, то присвоим этот список переменной y:



Затем вызовем метод plot и передадим ему в качестве параметров наши списки х и у и вызовем метод show() для отображения графика:

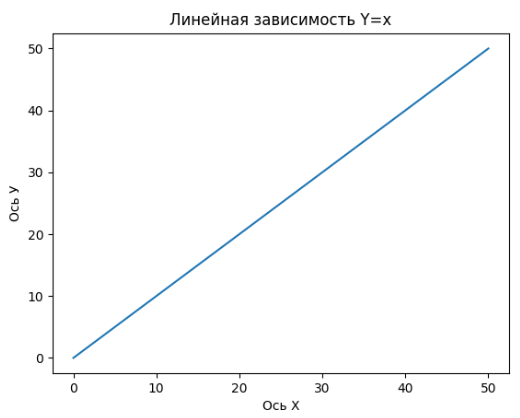
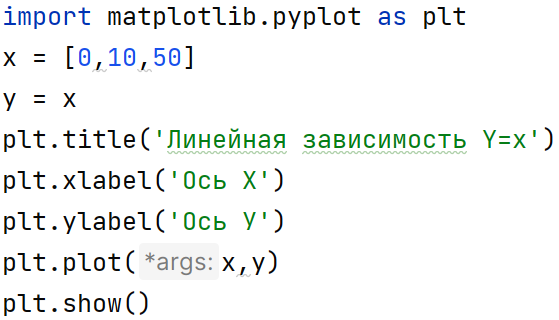


В результате мы получим:

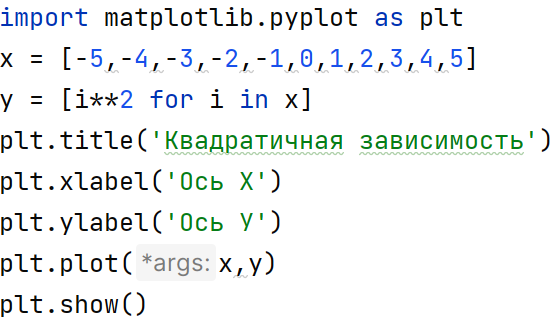


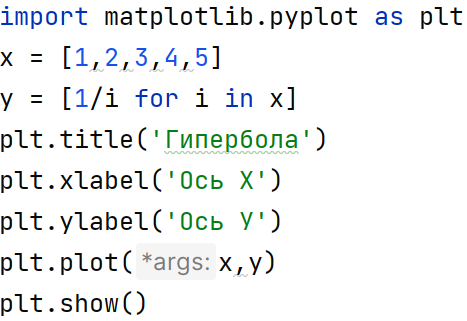
Обратите внимание, что деления на графике проставились автоматически.

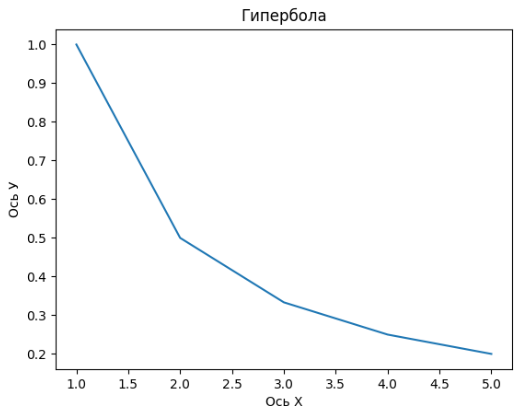
Добавим заголовок графика и подписи осей. Для этого изменим код:



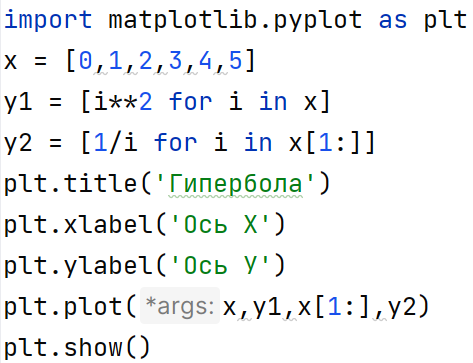
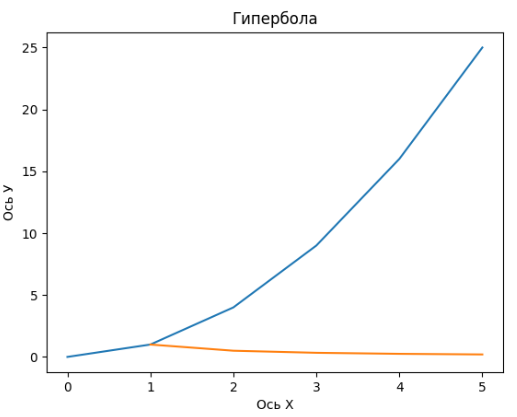
Задание 2. Постройте параболу (y=x2) и гиперболу (y=).







Как видите, принцип построение графиков одинаковый. Давайте отобразим оба графика на одном поле.

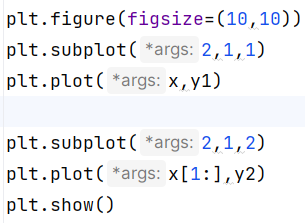


*Вопрос: почему в случае с гиперболой была использована запись «x[1:]»*

Как видно, в масштабе с параболой гипербола выглядит менее узнаваемо, кроме того, программа сама выделила цветом две разные функции.

Задание 3. Постройте предыдущие графики на разных полях.

Библиотека matplotlib позволяет выводить функции одновременно не только на одном поле, но на нескольких сразу. Изменим предыдущий код:



Здесь появились две новые функции: figure – определяет параметры всего поля отображения, в данном случае мы задали ему размер 10х10.

Функция subplot разбивает основное поле и принимает три параметра:

Первый – сколько вспомогательных полей будет отображаться;

Второй и третий указывает положение поля. Это можно представить следующим образом:

Поле 1,2

Поле 1,1

Все поле

В результате мы получим следующее:

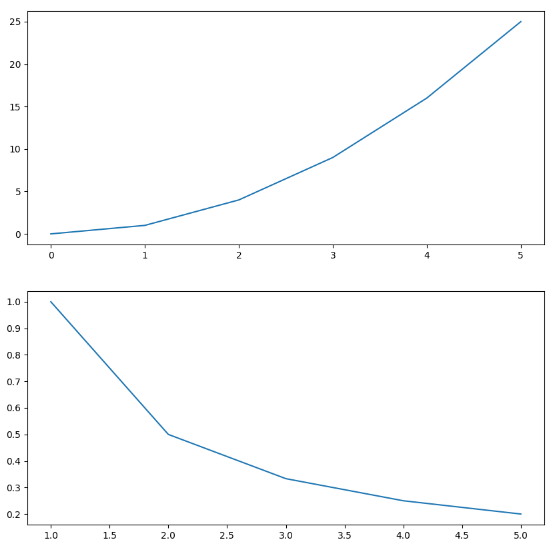




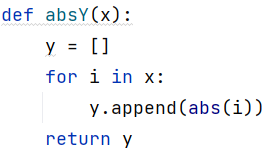
Рисунок 1 - основные элементы графика

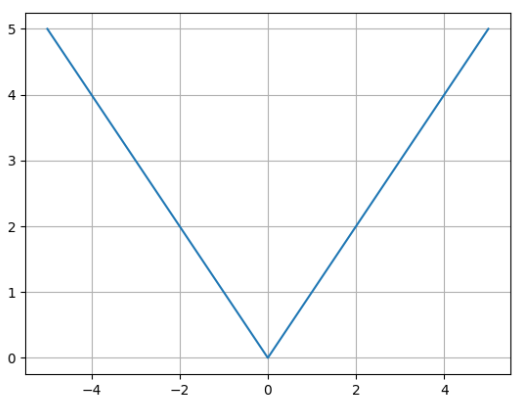
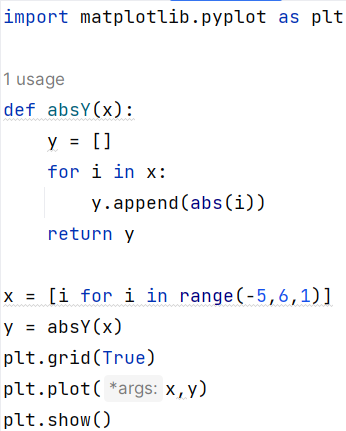
Задание 3: Постройте график функции y=|x|.

Как мы помним из курса математики, значения у такой функции выглядят следующим образом:

У =

Напишем функцию для нахождения значений У:

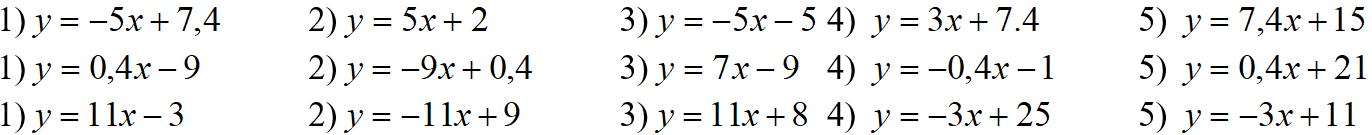


Затем зададим значения Х и У и отобразим график:

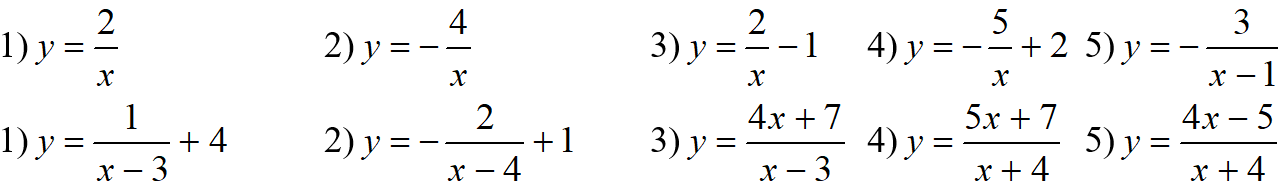
**Самостоятельная работа**

***Все задания помещаются в одно решение!***

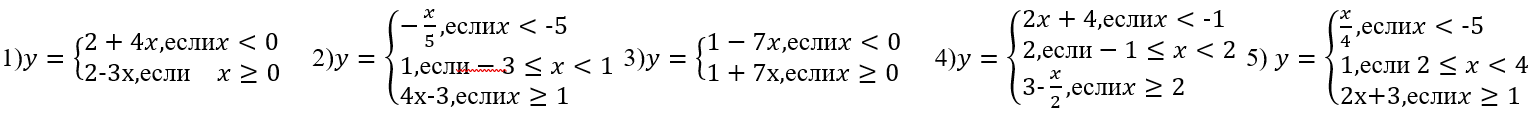
Построить график функции на одном поле, подписать легенду.



Построить график функции в одной программе на разных полях, подписать легенду.



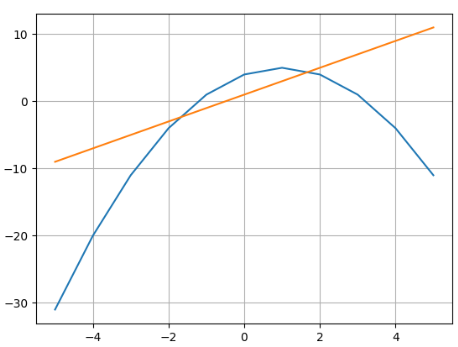
Построить график функции

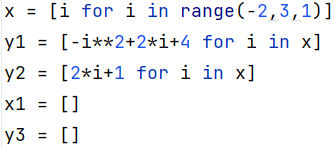


**Практическая работа №13**

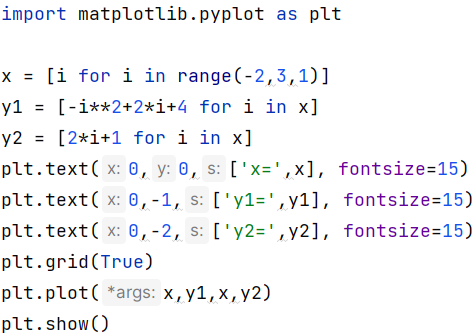
**Решение задач с использованием графиков**

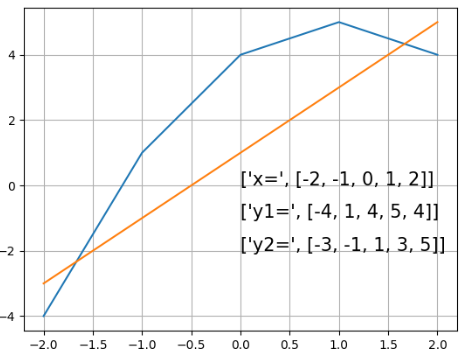
Задание 1: Заданы два уравнения: Y1 < -X2 + 2X + 4 и Y2 > 2X + 1. Найти количество точек с целыми координатами, которые лежат в области пересечения кривых.

Для решения этой задачи в первую очередь необходимо построить графики этих функций:



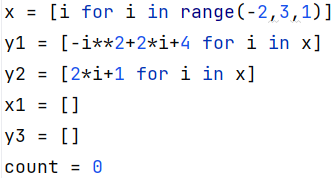
-5, 5,1

Значения Х можно брать любые. На графике видна область пересечения двух кривых (выделена зеленым цветом). Мы видим, что эта область лежит примерно между х=-2 и х=2. Давайте зададим такие значения Х, чтобы на графике осталась только область пересечения, и выведем на график координаты точек:



Теперь эту область видно более наглядно.

Создадим два пустых списка х1 и у1, в которые будем заносить координаты точек, удовлетворяющих условие задачи:



Давайте разберем какие точки нам необходимо проверить. По условию задачи те точки, которые лежат ниже функции Y1 (Y1 < -X2 + 2X + 4 и Y2 ) и выше функции Y2 (Y2 > 2X + 1). Рассмотрим таблицу с координатами точек:

|  |  |  |
| --- | --- | --- |
| **X** | **Y1** | **Y1** |
| *-2* | *-4* | *-3* |
| *-1* | *1* | *-1* |
| *0* | *4* | *1* |
| *1* | *5* | *3* |
| *2* | *4* | *5* |

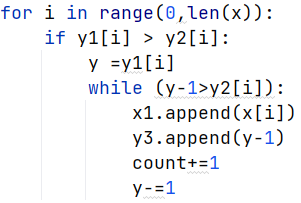
Нас интересуют значения где Y1>Y2. Выделим их в таблице цветом. Рассмотрим, какие значения Y удовлетворяют каждую пару (с учетом того, что точки должны лежать строго между двух функций):

(1,-1) – только 0 – 1>0>-1

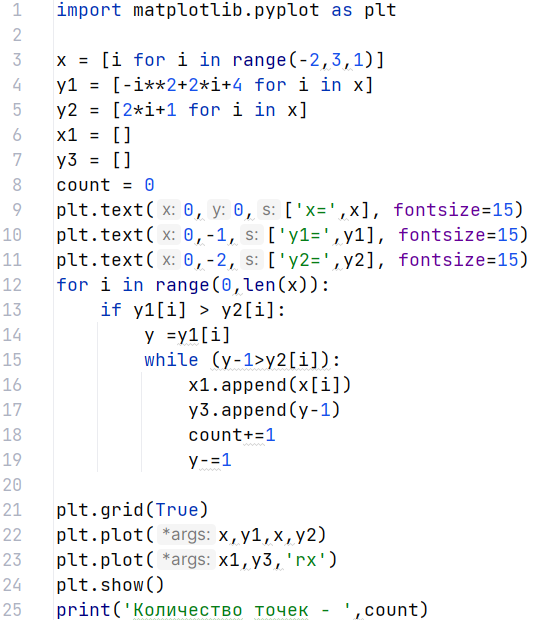
(4,1) – 3, 2 – 4>3, 2>1

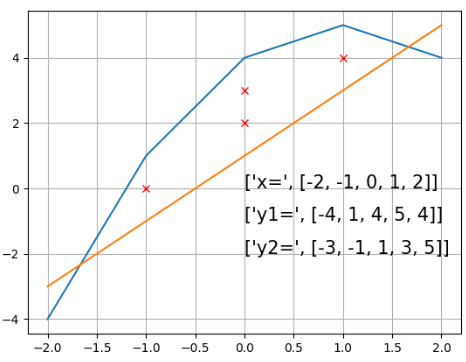
(5,3) – только 4 – 5>4>3

Исходя из этого, нам необходимо проверить все значения Y1 на условие Y1>Y2, если условие выполняется, то пока Y1-1 > Y2, записываем координаты точек.

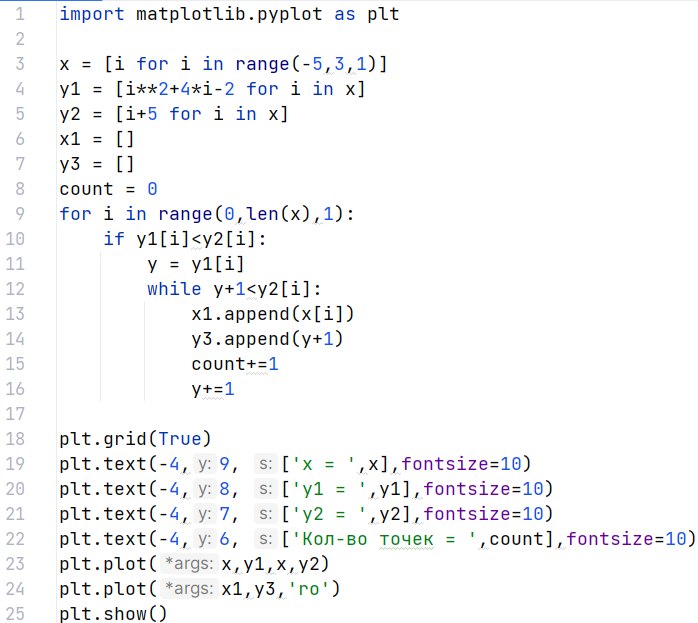


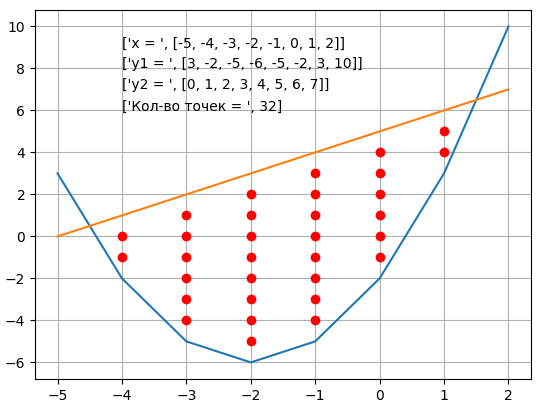
Осталось отобразить все это на графике:





Задание 2: Заданы два уравнения: Y1 > X2 + 4X – 2, Y2 < X + 5. Найти количество точек с целыми координатами, которые лежат в области пересечения кривых. Самостоятельно изучите предложенный код.





**Самостоятельная работа**

***Все задания помещаются в одно решение!***

|  |  |
| --- | --- |
| Задание № 1. Y1 > X2 + 2X + 3, Y2 < -X /4 + 10.  Задание № 2. Y1 > X2 + 2X + 4, Y2 < 2X + 12.  Задание № 3. Y1 > X2 – 2X + 5, Y2 < 3X + 11  Задание № 4. Y1 > X2 – 2X + 3, Y2 < –X/3 + 9.  Задание № 5. Y1 > X2 + 4X + 5, Y2 < –X/2 + 9.  Задание № 6. Y1 > X2 + 4X + 4, Y2 < 2X + 8.  Задание № 7. Y1 > X2 – 4X + 2, Y2 < –X + 6.  Задание № 8. Y1 > X2 – 4X + 5, Y2 < 3X + 8.  Задание № 9. Y1 < –X2 + 2X + 4 Y2 > X – 2.  Задание № 10. Y1 < –X2 + 2X + 3, Y2 > –2X – 3.  Задание № 11. Y1 < –X2 + 2X + 5, Y2 > 3X – 4.  Задание № 12. Y1 > –X2 + 4X + 2, Y2 > X/2 – 5.  Задание № 13. Y1 < –X2 + 4X + 5, Y2 > X – 3  Задание № 14. Y1 > X2 + 4X – 2, Y2 < -X /2 + 4.  Задание № 15. Y1 > X2 + 4X – 3, Y2 < 3X + 6.  Задание № 16. Y1 > X2 – 6X – 1, Y2 < 4X + 1  Задание № 17. Y1 > X2 – 6X – 3, Y2 < –X/2 + 8.  Задание № 18. Y1 > X2 + 4X – 1, Y2 < –X/3 + 9.  Задание № 19. Y1 > X2 + 4X + 2, Y2 < 2X + 10.  Задание № 20. Y1 > X2 – 2X + 3, Y2 < –X + 7.  Задание № 21. Y1 > X2 – 2X + 4, Y2 < 3X + 8.  Задание № 22. Y1 < –X2 + 2X + 2 Y2 > X – 2.  Задание № 23. Y1 < –X2 + 2X + 3, Y2 > –2X – 3.  Задание № 24. Y1 < –X2 + 4X + 5, Y2 > 3X – 4. | Задание № 25 Y1 < -X2 + 2X + 5, Y2 > -X /4 + 2  Задание № 26 Y1 < -X2 – 2X + 3, Y2 > 3X – 1  Задание № 27 Y1 < -X2 – 2X + 4, Y2 > –X/3 + 2.  Задание № 28 Y1 > -X2 + 4X + 5, Y2 > –X/2 – 3  Задание № 29 Y1 < -X2 + 4X + 4, Y2 > 2X – 2  Задание № 30 Y1 < -X2 – 4X + 2, Y2 > –X – 3  Задание № 31 Y1 < -X2 – 4X + 1, Y2 . 3X – 2  Задание № 32 Y1 > X2 + 2X + 4 Y2 < X + 5.  Задание № 33 Y1 > X2 + 2X + 3, Y2 < –2X + 4  Задание № 34 Y1 > X2 + 2X + 2, Y2 < 3X + 4.  Задание № 35 Y1 > X2 + 4X + 2, Y2 < X/2 + 5.  Задание № 36 Y1 > X2 + 4X + 5, Y2 < X – 3  Задание № 37 Y1 < -X2 + 4X – 2, Y2 > -X /2 – 1  Задание № 38 Y1 < -X2 + 4X – 3, Y2 > 3X – 2  Задание № 39 Y1 < -X2 – 2X – 1, Y2 > 4X – 3  Задание № 40 Y1 < -X2 – 6X – 3, Y2 > –X – 4  Задание № 41 Y1 > -X2 + 2X – 1, Y2 < –X – 3  Задание № 42 Y1 > X2 + 4X + 1, Y2 < 2X + 5.  Задание № 43 Y1 > X2 – 3X + 2, Y2 < –X + 7.  Задание № 44 Y1 > X2 – 2X + 4, Y2 < 2X + 5.  Задание № 45 Y1 < –X2 + 2X + 1 Y2 > X – 3  Задание № 46 Y1 < –X2 + 2X + 2, Y2 > –X – 3.  Задание № 47 Y1 < –X2 + 4X + 3, Y2 > X – 4.  Задание № 48. Y1 > –X2 + 4X + 2, Y2 > X/2 – 5. |

**Практическая работа №14**

**Построение столбчатых и круговых диаграмм**

Задание 1: Даны стоимости продуктов за прошлый год. Известно, что в этом году стоимость всех продуктов увеличилась на 15%. Постройте групповую столбчатую диаграмму стоимостей продуктов, отражающую стоимость за прошлый и текущий год.

В этой задаче кроме библиотеки MatPlotLib нам понадобится импортировать библиотеку numpy:

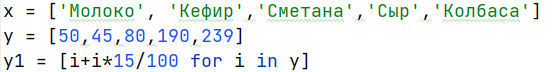


Создадим 3 списка:

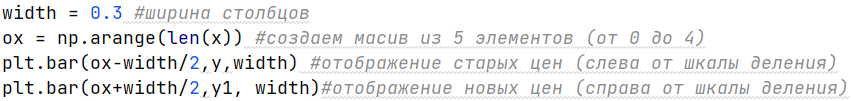
1 – названия наших продуктов

2 – стоимости в прошлом году

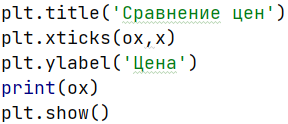
3 – стоимости в текущем году



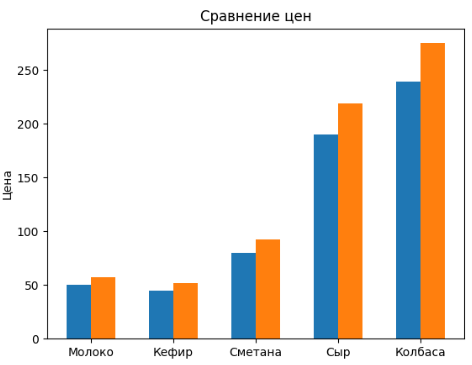
Чтобы отобразить столбцы со стоимостями одного товара, нам необходимо использовать следующие команды:



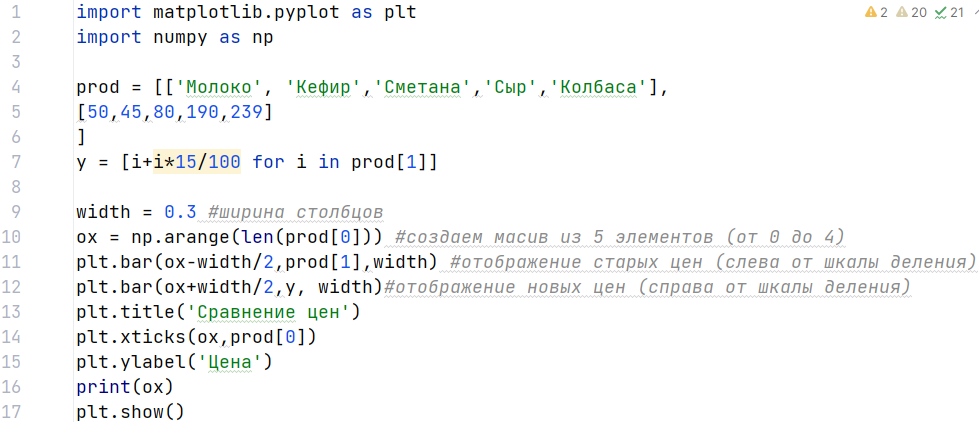
Осталось нанести отметки на график и отобразить его:



В результате получаем:



Давайте предположим, что продукты, и их старые цены хранятся в одном многомерном массиве. Давайте изменим код следующим образом:



Внесите изменения и проверьте, изменился ли результат?

Задание 2: В массиве хранится информация о погоде за месяц, где

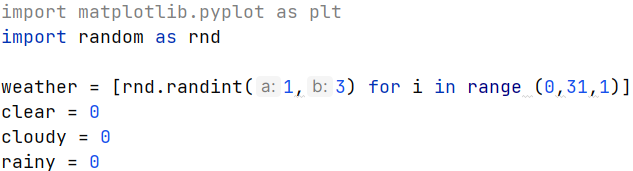
1 – ясно

2 – облачно

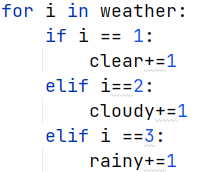
3 – дождь

Постройте круговую диаграмму, отражающую, какая погода преобладала в месяце.

Для начала давайте составим список с данными о погоде за каждый день и переменные, в которые занесем количество дней с каждой погодой:

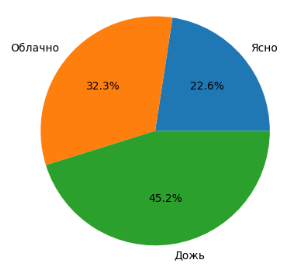


Далее рассчитаем количество дней определенной погоды:



Построим диаграмму:

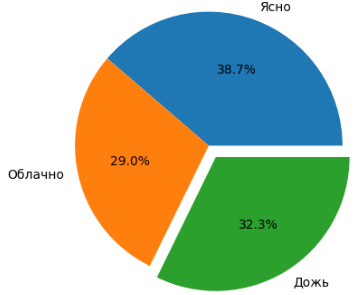




Обратите внимание, что при каждом запуске программы, данные будут разными, так как мы заполняем значения погоды случайными числами.

Давайте выделим сектор, показывающий количество дождливых дней, для этого дополним строчку:

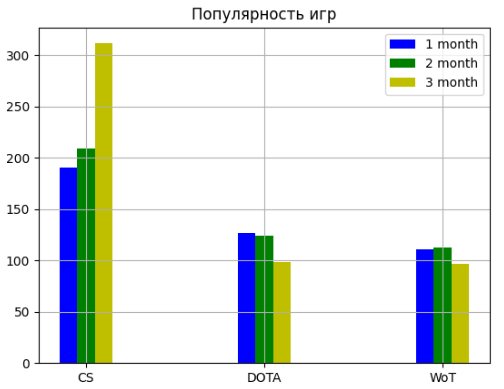




**Самостоятельная работа**

***Все задания помещаются в одно решение!***

1. В массиве приведены средние температуры за каждый месяц года. Определите среднюю температуру за каждое время года и постройте столбчатую диаграмму.
2. Определите 4 числовые характеристики, по которым пользователь будет сравнить товары. На основе введенных пользователем данных, постройте групповую столбчатую диаграмму сравнения характеристик двух товаров.
3. В массиве хранится информация о количестве игроков 3-х разных игр за первый месяц. Во втором и третьем месяце их количество увеличилось или уменьшилось на случайное количество процентов (используйте генератор случайных чисел). Постройте групповую столбчатую диаграмму для трех игр, отражающую количество игроков в каждый из трех месяцев. Примерный результат:



1. Дана стоимость некоторого товара. В течение года эта стоимость менялась в диапазоне от -5% до +5% каждый месяц. Постройте столбчатую диаграмму, отражающую стоимость товара за каждый месяц в течение года.
2. Известны годовые результаты учащихся по предмету. Постройте круговую диаграмму оценок, выделите сектор с оценкой «отлично».
3. Известная статистика посещаемости за месяц в 4 группах, в которой указаны:

-не пропустившие ни одного занятия

-пропустившие по уважительной причине

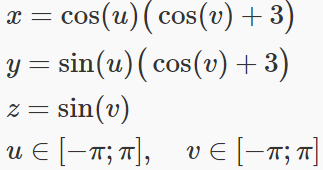
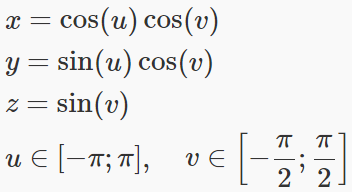
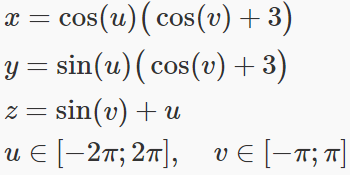
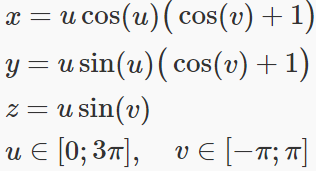
-прогулявшие занятия

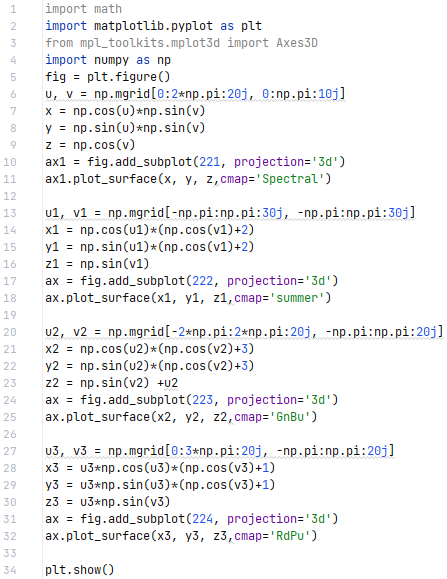
Постройте на одном поле четыре круговые диаграммы, отражающие посещаемость в четырех группах, выделите сектор с количеством прогулявших занятия.

**Практическая работа №15**

**Построение 3d графиков**

Задание: Постройте графики следующих функций:





Что у вас получилось? Постройте каждый график отдельно.