**Модуль 1. Объекты программы на Python. Основные алгоритмические конструкции**

**Тема 1. Знакомство с Python**

**Урок 3-5. Структура программы на Python, базовые типы данных, операторы ввода и вывода данных**

Задание 4

Давайте расширим память нашей программе-повторюшке – пусть она выслушает три наших утверждения (три строки), а потом эти три строки воспроизведёт. Напишите программу, которая принимает на ввод три строки, а затем выводит их в том же порядке.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| Привет!  Я твой создатель!  Теперь ты будешь мне служить! | Привет!  Я твой создатель!  Теперь ты будешь мне служить! |
|  |  |

a = input()  
b = input()  
c = input()  
print(a)  
print(b)  
print(c)

**Урок 6. Знакомство с IDE PyCharm. Установка и настройка среды программирования**

Задание 3

Напишите программу, которая для заданного отрезка вернёт информацию в следующей форме:

*Отрезок [* [левая граница] *;* [правая граница] *] имеет длину* [длина]

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| -5  7 | Отрезок [ -5 ; 7 ] имеет длину 12 |
| 2  3 | Отрезок [ 2 ; 3 ] имеет длину 1 |
|  |  |

a = input()  
a1 = int(a)  
b = input()  
b1 = int(b)  
c = b1 - a1  
print("Отрезок", "[", a, ";", b, "]", "имеет длину", c)

**Урок 7. Линейная программа. Построение диалога компьютер-пользователь**

Задание 6 (\*)

Геометрия берёт своё начало в Древнем Египте. Пусть эта задача будет данью колыбели геометрии. Вам необходимо написать программу, которая выводит треугольник (см. пример), построенный с помощью заданного символа, который вводится с клавиатуры.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| @ | @  @ @  @@@@@ |
| \* | \*  \* \*  \*\*\*\*\* |
|  |  |

a = input()  
b = ''  
print(b,b,a,b,b)  
print(b,a,a,b)  
print(a+a+a+a+a)

**Урок 9–10. Арифметические операции и выражения. Переменные и константы в выражениях. Стандартные функции. PEP8**

Задание 2

Предположим, что в каждой целочисленной точке числовой прямой расположена булочная. И один Юный Любитель Пирожков направляется из пункта A в пункт B. Если он окажется около булочной – он обязательно в неё зайдет и купит пирожок. Помогите ответить на вопрос – сколько пирожков съест Юный Любитель?

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 2  9 | 8 |
| 2  3 | 2 |
|  |  |

a = input()  
b = input()  
a1 = int(a)  
b1 = int(b)  
c = 1  
while a1 < b1:  
 c = c+1  
 a1 = a1 + 1  
print(c)

**Урок 11–13. Оператор присваивания. Решение вычислительных задач**

Задание 5

Конечно, Петя в лагере не только пил компот – ему приходилось в том числе и посещать учебные занятия. И вот на одном таком занятии Пете необходимо было построить график параболы, но высчитывать значение квадратичной функции в точке ему было лень – он написал программу.

Попробуйте восстановить программу, написанную Петей, если известно, что она высчитывала значение функции в произвольной точке:

*.*

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 0 | 7 |
| 2 | 21 |
|  |  |

x = int(input())  
y = (5 \* x \* x) - (3 \* x) + 7  
print(y)

**Тема 2. Основы программирования в среде Python**

**Задания к уроку 15-16. Линейная программа. Регулирование вывода через sep и end**

Задание 2

Напишите программу, которая хвалит ученика, а похвала при этом описывается с помощью однородных определений (эпитетов) в следующей форме:

**«Ученик Василий — <эпитет\_1>, <эпитет\_2>, <эпитет\_3>.»**

В качестве входных данных программа принимает три строки — эпитеты для характеристики ученика Василия.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| умный красивый добрый | Ученик Василий — умный, красивый, добрый. |
| двоечник бездельник хулиган | Ученик Василий — двоечник, бездельник, хулиган. |
|  |  |

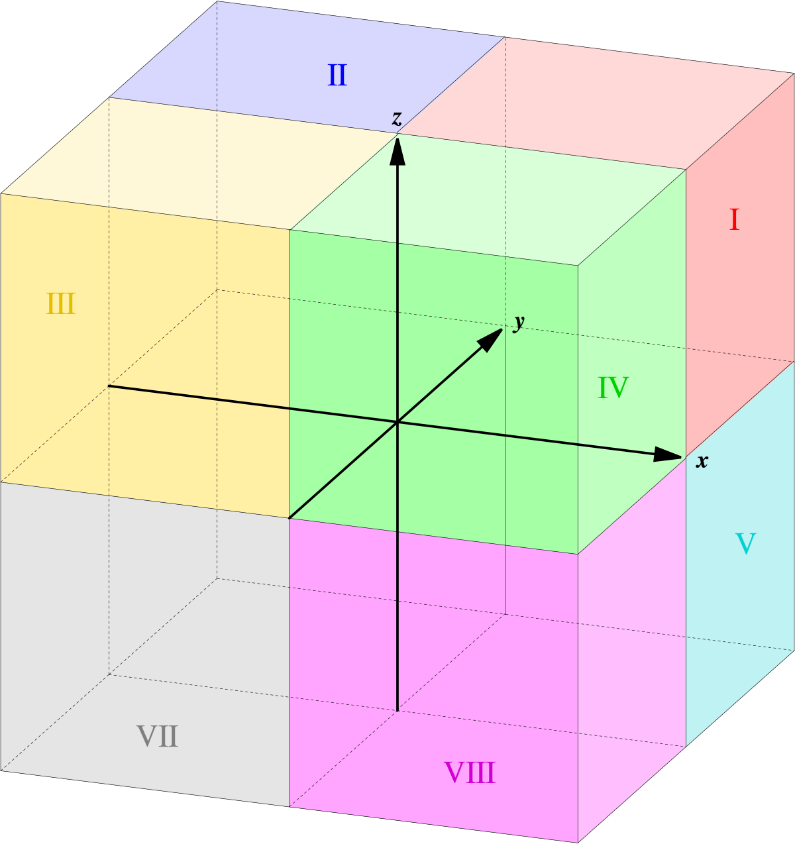
a = input()  
b = input()  
c = input()  
print("Ученик Василий ",end='- ')  
print(a,b,c, sep=', ' ,end='.')

**Урок 20-21. Логические операции в сложных условиях. Программирование ветвления, отладка кода**

Задание 5

Вам необходимо написать программу, которая для координат конкретной точки в трёхмерном пространстве, утверждает, в каком октанте она находится.

Расположение октантов можно посмотреть на картинке ниже:



|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 2 3 2 | 1 |
| 2 3 -2 | 5 |
|  |  |

initialData = input()  
data = initialData.split()  
x = float(data[0])  
y = float(data[1])  
z = float(data[2])  
if (x > 0) and (y > 0) and (z > 0):  
 print("1")  
elif (x < 0) and (y > 0) and (z > 0):  
 print("2")  
elif (x < 0) and (y < 0) and (z > 0):  
 print("3")  
elif (x > 0) and (y < 0) and (z > 0):  
 print("4")  
elif (x > 0) and (y > 0) and (z < 0):  
 print("5")  
elif (x < 0) and (y > 0) and (z < 0):  
 print("6")  
elif (x < 0) and (y < 0) and (z < 0):  
 print("7")  
elif (x > 0) and (y < 0) and (z < 0):  
 print("8")

**Урок 22-23. Вложенность в структурах ветвления. Применение оператора выбора**

Задание 1

Используя только условный оператор и оператор присваивания (не считая ввода и вывода), напишите программу, которая для двух введённых целых чисел, выводит их в порядке неубывания.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 2 3 | 2 3 |
| 3 2 | 2 3 |
|  |  |

a = int(input())  
b = int(input())  
if a > b:  
 print(b,a)  
else:  
 print(a,b)

**Тема 3. Реализация алгоритмов ветвления в среде Python**

**Урок 31-32. Вложенность циклов. Тестирование программы**

Задание 1

В качестве разминки всегда стоит брать задачу простую – например текущую задачу.

Вам необходимо написать программу, которая принимает на вход два натуральных числа n и m, после чего программа в n строках выводит первые m натуральных чисел.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 2  3 | 1 2 3  1 2 3 |
|  |  |

n = int(input("Введите количество строк: "))  
m = int(input("Введите количество чисел в строке: "))  
  
for i in range(n):  
 for j in range(1, m + 1):  
 print(j, end=" ")  
 print()

**Урок 33-34. Циклы. Классические алгоритмы: алгоритм Евклида, выделение и обработка цифр числа, вычисление факториала, получение делителей натурального числа**

Задание 3

Любой юный исследователь хочет оптимизировать свою деятельность. И вот перед Вами, как перед юным исследователем стоит задача – определить количество цифр в числе, которые являются нечётными.

Вам необходимо написать программу, которая на вход принимает некоторое натуральное число a и возвращает целое неотрицательное число – количество нечётных цифр в записи исходного число a.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 123 | 2 |
|  |  |

n = int(input())  
  
count = 0  
while n != 0:  
 c = n % 10  
 if c % 2 == 1:  
 count += 1  
 n = n // 10  
  
print(count)