

## Занятие 3

### Цикл for

Одно из преимуществ компьютеров перед людьми - способность повторять одни и те же действия многократно, быстро и совсем не утомляясь.

Цикл for замечательно работает, если мы заранее знаем, сколько повторений (итераций) нам требуется сделать. Однократное выполнение тела цикла называется **итерацией цикла**.

Рассмотрим следующий код:

```
for i in range(10):  
    print(i)
```

Результатом выполнения такого кода будет:

```
0 1 2 3 4 5 6 7 8 9
```

Функцию **range()** мы можем генерировать любую последовательность целых чисел с шагом 1. Но, что делать если нужно поменять шаг? Как быть если мы хотим сгенерировать последовательность чисел 5, 10, 15, 20, 25? В этом случае нужно использовать еще один параметр в функции **range()**:

```
range(n, m, k).
```

Первый параметр **n** задает **старт последовательности**

Второй параметр **m** задает **стоп последовательности**

Третий **k** – **шаг генерации** чисел.

## Задачи

1. Напишите программу, которая выводит слово «Привет!» (без кавычек) 10 раз.

**Формат выходных данных:**

Программа должна вывести 10 раз текст «Привет!», каждый на отдельной строке.

2. Дано предложение и количество раз которое его надо повторить. Напишите программу, которая повторяет данное предложение нужное количество раз.

**Формат входных данных:**

В первой строке записано текстовое предложение, во второй — количество повторений.

**Формат выходных данных:**

Программа должна вывести указанное текстовое предложение нужное количество раз. Каждое повторение должно начинаться с новой строки.

3. На вход программе подается натуральное число  $n$ . Напишите программу, которая печатает звездный прямоугольник размерами  $n \times 10$ .

**Формат входных данных:**

На вход программе подаётся натуральное число  $n$  — высота звездного прямоугольника.

**Формат выходных данных:**

Программа должна вывести звездный прямоугольник размерами  $n \times 10$ .

**Подсказка.** Для печати звездной линии используйте умножение строки на число.

4. На вход программе подается натуральное число  $n$ . Напишите программу, которая печатает звездный треугольник.

**Формат входных данных:**

На вход программе подаётся натуральное число  $n$  — высота звездного треугольника.

**Формат выходных данных:**

Программа должна вывести звездный треугольник высотой  $n$ .

**Пример:**

$n = 4$

\*

\*\*

\*\*\*

\*\*\*\*

5. На вход программе подается натуральное число  $n$ . Напишите программу, которая для каждого из чисел от 0 до  $n$  (включительно) выводит фразу: «Квадрат числа [число] равен [число]» (без кавычек).

**Формат входных данных:**

На вход программе подается натуральное число  $n$ .

**Формат выходных данных:**

Программа должна вывести текст в соответствии с условием задачи.

6. Даны два целых числа  $m$  и  $n$ . Напишите программу, которая выводит все числа от  $m$  до  $n$  включительно в порядке возрастания, если  $m < n$ , или в порядке убывания в противном случае.

**Формат входных данных:**

На вход программе подаются два целых числа  $m$  и  $n$ , каждое на отдельной строке.

**Формат выходных данных:**

Программа должна вывести числа в соответствии с условием задачи.

7. Дано натуральное число  $n$ . Напишите программу, которая выводит таблицу умножения на  $n$ .

**Формат входных данных:**

На вход программе подается натуральное число.

**Формат выходных данных:**

Программа должна вывести таблицу умножения на введенное число.

**Примечание.** В качестве знака умножения используйте английскую букву  $x$ .

8. На вход программе подаются два целых числа  $a$  и  $b$  ( $a \leq b$ ). Напишите программу, которая подсчитывает количество чисел в диапазоне от  $a$  до  $b$  включительно, куб которых оканчивается на 4 или 9.

**Формат входных данных:**

На вход программе подаются два целых числа  $a$  и  $b$  ( $a \leq b$ ).

**Формат выходных данных:**

Программа должна вывести одно целое число в соответствии с условием программы.

**Примечание.** 1) Куб числа  $a$  – это его третья степень  $a^{**3}$ .

2) Для подсчёта количества будет полезно использовать вспомогательную переменную.