

# Рекурсия

1. Написать функцию, которая принимает в качестве параметра натуральное число и с использованием рекурсии возвращает сумму чисел от 1 до принимаемого числа.

Пример работы функции:

**func(5)** // 15, т.к.  $1 + 2 + 3 + 4 + 5 = 15$

2. Написать функцию, которая принимает в качестве параметра натуральное число и с использованием рекурсии возвращает факториал числа.

Пример работы функции:

**func(5)** // 120, т.к.  $5! = 1 * 2 * 3 * 4 * 5 = 120$

3. Написать функцию, определяющую сумму чисел **x** и **y** (**x** и **y** – целые; **x** > 0, **y** > 0) рекурсивно. Складывать **x** и **y** нельзя.

Пример работы функции:

**func(12, 21)** // 33

4. Сгенерировать массив из 10 случайных целых чисел в диапазоне [1, 9]. Написать функцию, которая принимает в качестве параметра этот массив и с использованием рекурсии выводит элементы массива на экран последовательно (по одному элементу).

5. Написать функцию, которая осуществляет поиск наибольшего значения в массиве целых чисел рекурсивно.

6. Написать функцию, которая принимает в качестве параметра натуральное число. До тех пор, пока сумма цифр числа не станет однозначной, функция должна складывать цифры этого числа с использованием рекурсии. Функция должна вернуть однозначное число, получившееся после сложения цифр принимаемого числа.

Пример работы функции:

1) **func(8)** // 8, 8 – однозначное число

2) **func(25)** // 7, 25 – двузначное число,  $2 + 5 = 7$  (7 – однозначное число)

3) **func(123456)** // 3, 123456 – 6-значное число,  $1 + 2 + 3 + 4 + 5 + 6 = 21$  (21 – двузначное число),  $2 + 1 = 3$  (3 – однозначное число)