

# ДЕКЛАРАТИВНОЕ ПРОГРАММИРОВАНИЕ

---

Рекурсия

# Факториал

$$n! = \begin{cases} 1 & n = 0, \\ n \cdot (n - 1)! & n > 0. \end{cases}$$

# Факториал (с ошибкой!)

$$n! = \begin{cases} 1 & n = 0, \\ n \cdot (n - 1)! & n > 0. \end{cases}$$

```
factorial n = n * factorial (n - 1)
```

# Факториал

$$n! = \begin{cases} 1 & n = 0, \\ n \cdot (n - 1)! & n > 0. \end{cases}$$

`factorial 0 = 1`

`factorial n = n * factorial (n - 1)`

# Правила

- (1) определите конечную цель (или цели);
- (2) определите, что происходит, когда цель достигнута;
- (3) перечислите все имеющиеся возможности;
- (4) определите, какие действия следует повторять;
- (5) убедитесь, что каждая из возможностей приближает к цели.

# Факториал

$$n! = \begin{cases} 1 & n = 0, \\ n \cdot (n - 1)! & n > 0. \end{cases}$$

`factorial 0 = 1`

`factorial n = n * factorial (n - 1)`

Что произойдет при вызове `factorial (-1)` ?

# Упражнение

Определите функцию, вычисляющую двойной факториал, то есть произведение натуральных чисел, не превосходящих заданного числа и имеющих ту же четность.

Например:

$$7!! = 7 \cdot 5 \cdot 3 \cdot 1,$$

$$8!! = 8 \cdot 6 \cdot 4 \cdot 2.$$

Предполагается, что аргумент функции может принимать только неотрицательные значения.

# Упражнение: Функция Аккермана

$A(m, n) =$

- $n + 1$ , если  $m = 0$
- $A(m - 1, 1)$ , если  $m > 0, n = 0$
- $A(m - 1, A(m, n - 1))$ , если  $m > 0, n > 0$

Эта функция растёт очень быстро, например, число  $A(4, 4)$  настолько велико, что количество цифр в порядке этого числа многократно превосходит количество атомов в наблюдаемой части Вселенной.

[Функция Аккермана \(wiki\)](#)



# Упражнение

1. Напишите функцию, вычисляющую n-е число Фибоначчи

0, 1, 1, 2, 3, 5, 8,...

$$F_0=0$$

$$F_1=1$$

$$F_n=F_{n-2}+F_{n-1}$$

2. Дополните реализацию функции, чтобы она работала и для отрицательных чисел. Рекуррентное соотношение то же!

:set +s

# Упражнения

Напишите функцию вычисляющую сумму цифр числа

Напишите функцию  
«переворачивающую» число: 1234 -> 4321

# Упражнение

Напишите функцию, которая осуществляет обратный отсчет для ракеты! Для данного числа функция должна создать строку, которая:

- начинается с "Ready!"
- считает от заданного числа до единицы
- заканчивается "Liftoff!".

Например,

```
>countdown 4
```

```
"Ready! 4... 3... 2... 1... Liftoff!"
```

Подсказки:

- объединить строки можно с помощью оператора ++
- функция `show` может преобразовать число в строку

# Упражнение\*

Какое максимальное количество пересечений могут образовать  $n$  непараллельных прямых?

