

# Семинар 5

## Задача 1

Напишите функцию `square()`, которая принимает на вход число и:

- возвращает его квадрат;
- выводит на экран сообщение вида "Квадрат числа равен: n" и ничего не возвращает;
- выводит на экран сообщение вида "Квадрат числа равен: n" и возвращает квадрат числа;

## Задача 2

Напишите функцию `nims()`, которая принимает на вход целое число, а возвращает список из двух элементов: целое число, предшествующее принятому на вход, и число, следующее за ним.

**Пример:**

*Входные данные:*

7

*Выходные данные:*

[6, 8]

## Задача 3

Напишите функцию `str_lower()`, которая принимает на вход строку (набор слов через пробел), а возвращает список ее элементов в нижнем регистре.

**Пример:**

*Входные данные:*

"В лесу родилась ёлочка В лесу она росла"

*Выходные данные:*

['в', 'лесу', 'родилась', 'ёлочка', 'в', 'лесу', 'она', 'росла']

## Задача 4

Напишите функцию `my_log()`, которая принимает на вход список чисел, и возвращает список их натуральных логарифмов. Если число меньше или равно 0, на его месте в возвращаемом списке должно быть `None`.

**Пример:**

*Входные данные:*

```
[1, 3, 2.5, -1, 9, 0, 2.71]
```

*Выходные данные:*

```
[0.0,  
 1.0986122886681098,  
 0.9162907318741551,  
 None,  
 2.1972245773362196,  
 None,  
 0.9969486348916096]
```

## Задача 5

Напишите функцию, которая принимает на вход два списка (имена людей и их возраст) и делает следующее:

- если эти два списка имеют одинаковую длину, то возвращает словарь с парами имя-возраст;
- если эти два списка имеют разную длину, то функция выводит на экран сообщение "Списки имеют разную длину" и возвращает пустой словарь.

**Пример:**

*Входные данные:*

```
["Ann", "Tim", "Sam"]  
[12, 23, 17]
```

*Выходные данные:*

```
{"Ann": 12, "Tim": 23, "Sam": 17}
```

*Входные данные:*

```
["Ann", "Tim", "Sam"]  
[12, 23, 17, 45]
```

*Выходные данные:*

```
Списки имеют разную длину  
{}
```

## Задача 6

Напишите функцию `binom_prob()`, которая считает вероятность того, что проведя  $n$  испытаний Бернулли с вероятностью успеха  $p$ , мы получим ровно  $k$  успехов. Эта функция принимает на вход три аргумента:  $p$ ,  $n$  и  $k$  и возвращает такой результат:

$$P(X = k) = C_n^k \cdot p^k \cdot (1 - p)^{n-k}$$

Для расчета биномиального коэффициента можно использовать функцию, написанную на лекции.

## Задача 7

Напишите функцию `all_sort()` , которая принимает на вход перечень целых чисел через запятую (не список, а просто сколько угодно значений через запятую) и возвращает список этих значений, отсортированный по возрастанию.

### Пример:

*Входные данные:*

7, 6, 1, 3, 8, 0, -2

*Выходные данные:*

[-2, 0, 1, 3, 6, 7, 8]