

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития  
Кафедра инфокоммуникаций

**«Функции с переменным числом параметров в Python»**

**ОТЧЕТ**  
**по лабораторной работе №13**  
**дисциплины**  
**«Основы программной инженерии»**

Выполнил:

Борсуков Владислав Олегович  
2 курс, группа ПИЖ-б-о-21-1,  
09.03.04 «Программная  
инженерия», направленность  
(профиль) «Разработка и  
сопровождение программного  
обеспечения», очная форма  
обучения

---

(подпись)

Проверил:

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2022 г.

### Проработка примера:

```
1 ▶ #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def median(*args):
5     if args:
6         values = [float(arg) for arg in args]
7         values.sort()
8         n = len(values)
9         idx = n // 2
10        if n % 2:
11            return values[idx]
12        else:
13            return (values[idx - 1] + values[idx]) / 2
14    else:
15        return None
16
17
18 ▶ if __name__ == "__main__":
19     print(median())
20     print(median(3, 7, 1, 6, 9))
21     print(median(1, 5, 8, 4, 3, 9))
```

median() > if args

Run: 1\_ex ×

F:\GitLab\Lab-2.11\venv\Scripts\python.exe F:\GitLab\Lab-2.10\PyCharm\1\_ex.py

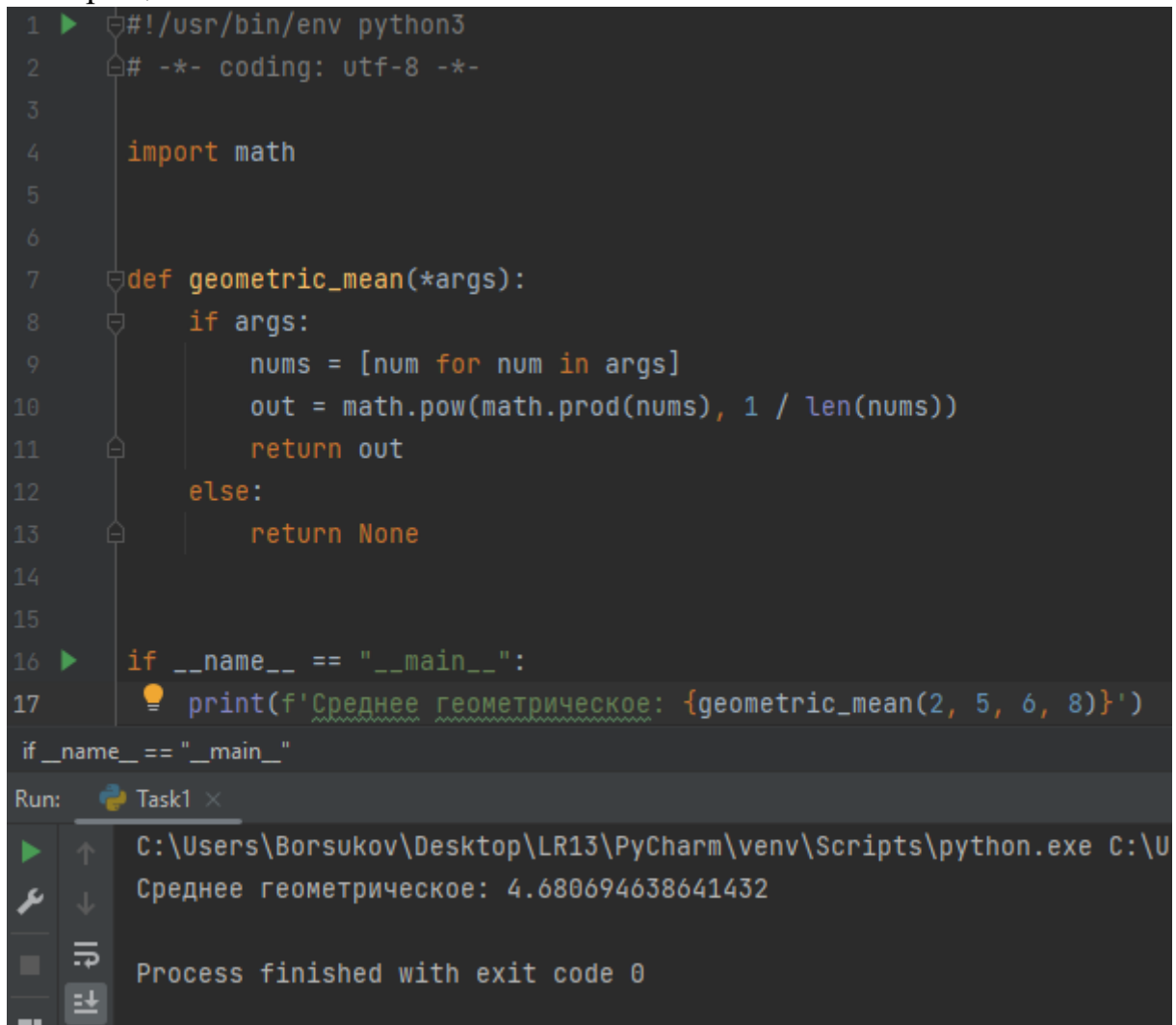
None  
6.0  
4.5

Рисунок 1 – Код и результат работы примера

**Задание №1:** написать функцию, вычисляющую среднее геометрическое своих аргументов  $a_1, a_2, \dots, a_n$

$$G = \sqrt[n]{\prod_{k=1}^n a_k}.$$

Если функции передается пустой список аргументов, то она должна возвращать значение None.



```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import math
5
6
7  def geometric_mean(*args):
8      if args:
9          nums = [num for num in args]
10         out = math.pow(math.prod(nums), 1 / len(nums))
11         return out
12     else:
13         return None
14
15
16 if __name__ == "__main__":
17     print(f'Среднее геометрическое: {geometric_mean(2, 5, 6, 8)}')
```

Run: Task1 ×

C:\Users\Borsukov\Desktop\LR13\PyCharm\venv\Scripts\python.exe C:\U  
Среднее геометрическое: 4.680694638641432

Process finished with exit code 0

Рисунок 2 – Код и результат работы программы задания №1

**Задание №2:** решить поставленную задачу: написать функцию, вычисляющую среднее гармоническое своих аргументов  $a_1, a_2, \dots, a_n$

$$\frac{n}{H} = \sum_{k=1}^n \frac{1}{a_k}.$$

Если функции передается пустой список аргументов, то она должна возвращать значение None.

```
1 ▶ #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 import math
5
6
7 def average_harm(*args):
8     if args:
9         nums = [1/num for num in args]
10        out = len(nums)/math.fsum(nums)
11        return out
12    else:
13        return None
14
15
16 ▶ if __name__ == "__main__":
17     print(f'Среднее гармоническое: {average_harm(5, 2, 3)}')
```

average\_harm() > if args

Run: Task2 ×

C:\Users\Borsukov\Desktop\LR13\PyCharm\venv\Scripts\python.exe  
Среднее гармоническое: 2.9032258064516134

Process finished with exit code 0

Рисунок 3 – Код и результат работы программы задания №2

**Индивидуальное задание:** напишите функцию, принимающую произвольное количество аргументов, и возвращающую требуемое значение. Если функции передается пустой список аргументов, то она должна возвращать значение None. Номер варианта определяется по согласованию с преподавателем. В процессе решения не использовать преобразования конструкции `*args` в список или иную структуру данных.

Произведение аргументов, расположенных между максимальным и минимальным аргументами

```
1  ▶  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import math
5
6
7  def multiply_new(*args):
8      if args:
9          min_i = args.index(min(args)) + 1
10         max_i = args.index(max(args))
11         return math.prod(args[min_i:max_i])
12     else:
13         return None
14
15
16  ▶  if __name__ == "__main__":
17      print(f'Произведение чисел между макс и мин: {multiply_new(3,6,7,9)}')
```

Run: Individual ×

▶ ↑ C:\Users\Borsukov\Desktop\LR13\PyCharm\venv\Scripts\python.exe C:\Users\Bo  
⚙ ↓ Произведение чисел между макс и мин: 42

▢ ↺ Process finished with exit code 0

⌵ ↻

Рисунок 4 – Код и результат работы программы индивидуально задания

## Контрольные вопросы

### 1. Какие аргументы называются позиционными в Python?

Позиционные это аргументы вызванные при помощи оператора \*

### 2. Какие аргументы называются именованными в Python?

Именованные это аргументы вызванные при помощи оператора \*\*

### 3. Для чего используется оператор \* ?

Этот оператор позволяет «распаковывать» объекты, внутри которых хранятся некие элементы. Вот пример:

```
a = [1,2,3]
b = [*a,4,5,6]
print(b) # [1,2,3,4,5,6]
```

Тут берётся содержимое списка `a`, распаковывается, и помещается в список `b`.

### 4. Каково назначение конструкций \*args и \*\*kwargs ?

```
def printScores(student, *scores):
    print(f"Student Name: {student}")
    for score in scores:
        print(score)
printScores("Jonathan",100, 95, 88, 92, 99)
"""
Student Name: Jonathan
100
95
88
92
99
"""
```

```
def printPetNames(owner, **pets):
    print(f"Owner Name: {owner}")
    for pet,name in pets.items():
        print(f"{pet}: {name}")
printPetNames("Jonathan", dog="Brock", fish=["Larry", "Curly", "Moe"], turtle="Shelldon")
"""
Owner Name: Jonathan
dog: Brock
fish: ['Larry', 'Curly', 'Moe']
turtle: Shelldon
"""
```