

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Основы кроссплатформенного программирования  
Отчет по лабораторной работе №5**

**Функции с переменным числом параметров в Python**

Выполнила студентка группы  
ИТС-б-о-20-1 (2)

Скачедубова А.В « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверил к.т.н., доцент

Кафедры инфокоммуникаций

Воронкин Р.А.

---

(подпись)

**Цель работы:** приобрести навыки по работе с функциями с переменным числом параметров при написании программ с помощью языка программирования Python.

**Ссылка на репозиторий:** [http://github.com/Any3002/lab\\_5](http://github.com/Any3002/lab_5)

**Порядок выполнения работы:**

1. Изучила теоретический материал работы.
2. Создала общедоступный репозиторий на GitHub.
3. Проработала примеры лабораторной работы.

Пример №1:

```
"C:\Users\Aspire 3\PycharmProjects\py
1 is stored in a
2 is stored in b
3 is stored in c

Process finished with exit code 0
```

Рисунок 1 – Отработанный пример №1

Пример №2. Для вызова функции необходимы все три аргумента. Если пропустить хотя бы один из них—будет выдано сообщение об ошибке:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def print_these(a, b, c):
    print(a, "is stored in a")
    print(b, "is stored in b")
    print(c, "is stored in c")

if __name__ == '__main__':
    print_these(1, 2)
```

Рисунок 2 – Код отработанного примера №2

```
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe"
Traceback (most recent call last):
  File "C:\Users\Aspire 3\PycharmProjects\pythonProject4\primer_2.py", line 1, in <module>
    print_these(1, 2)
TypeError: print_these() missing 1 required positional argument: 'c'

Process finished with exit code 1
```

Рисунок 3 – Отработанный пример №2

Пример №3:

```
primer_3 x
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe"
1 is stored in a
2 is stored in b
None is stored in c
Process finished with exit code 0
```

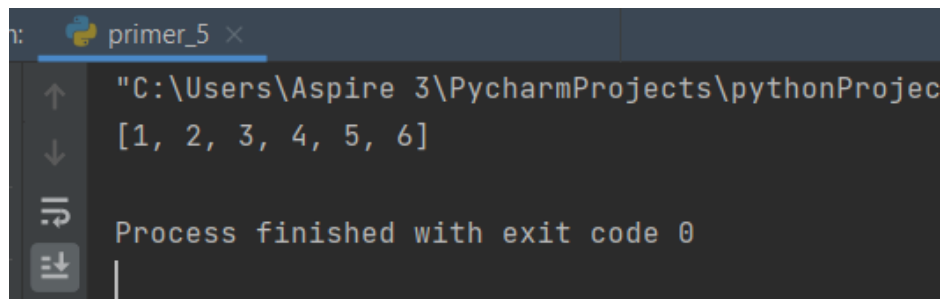
Рисунок 4 – Отработанный пример №3

Пример №4:

```
primer_4 x
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe"
1 is stored in a
None is stored in b
3 is stored in c
Process finished with exit code 0
```

Рисунок 5 – Отработанный пример №4

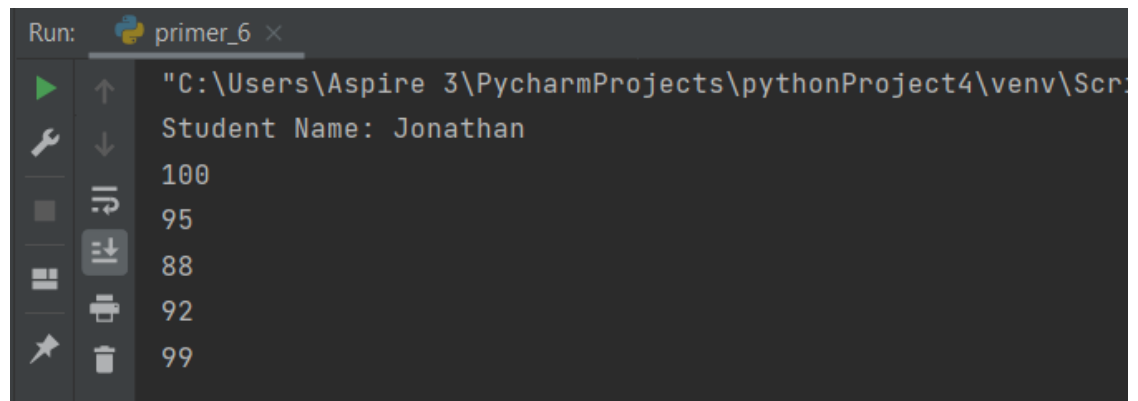
Пример №5:



```
Run: primer_5 x
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe"
[1, 2, 3, 4, 5, 6]
Process finished with exit code 0
```

Рисунок 6 – Отработанный пример №5

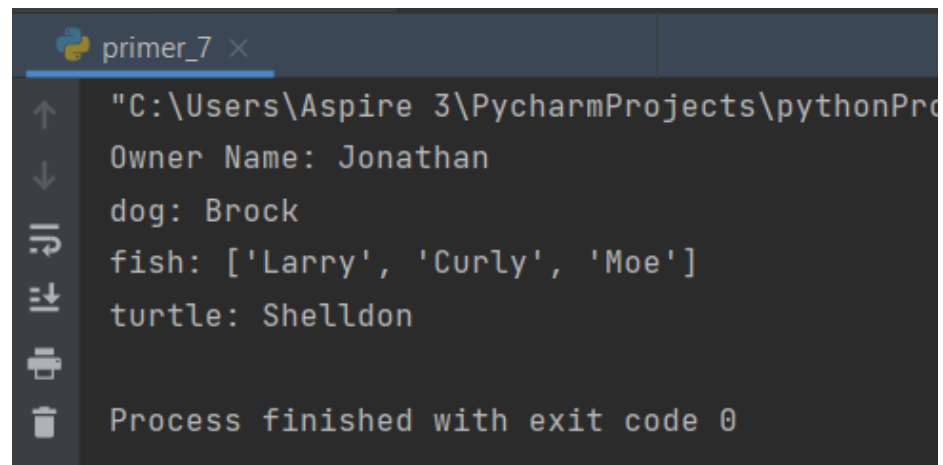
Пример №6:



```
Run: primer_6 x
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe"
Student Name: Jonathan
100
95
88
92
99
Process finished with exit code 0
```

Рисунок 7 – Отработанный пример №6

Пример №7:



```
Run: primer_7 x
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe"
Owner Name: Jonathan
dog: Brock
fish: ['Larry', 'Curly', 'Moe']
turtle: Shelldon
Process finished with exit code 0
```

Рисунок 8– Отработанный пример №7

Пример №8: разработать функцию для определения медианы значений аргументов функции. Если функции передается пустой список аргументов, то она должна возвращать значение None:

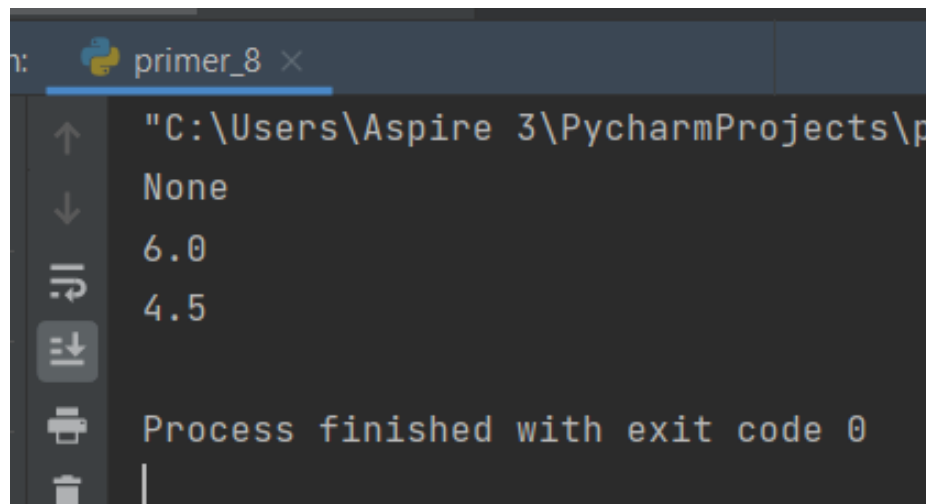


Рисунок 9 – Отработанный пример №8

4. Решила поставленную задачу:

8. Решить поставленную задачу: написать функцию, вычисляющую среднее геометрическое своих аргументов  $a_1, a_2, \dots, a_n$

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

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

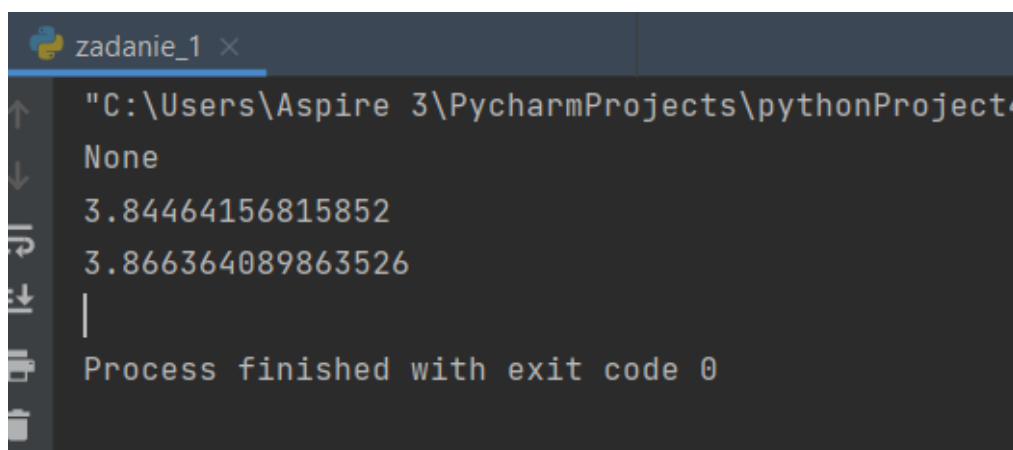


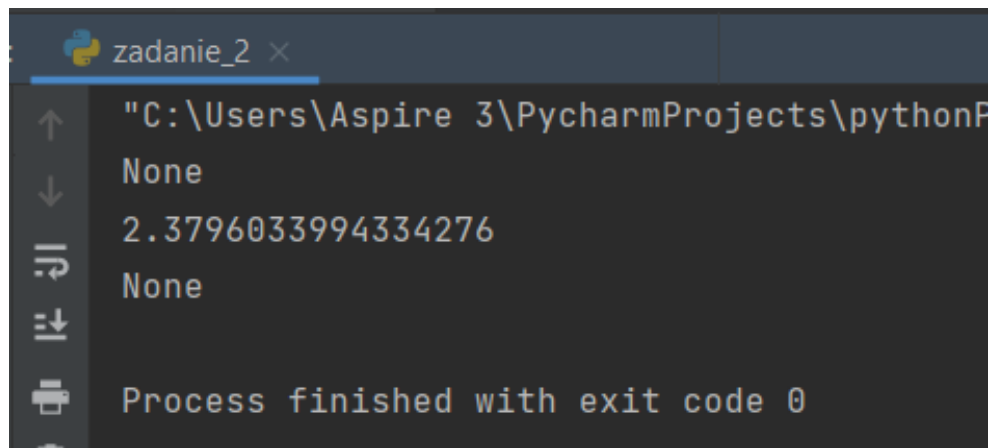
Рисунок 10 – Результат задачи №1

5. Решила поставленную задачу №2:

9. Решить поставленную задачу: написать функцию, вычисляющую среднее гармоническое своих аргументов  $a_1, a_2, \dots, a_n$

$$\frac{n}{H} = \sum_{k=1}^n \frac{1}{a_k}. \quad (2)$$

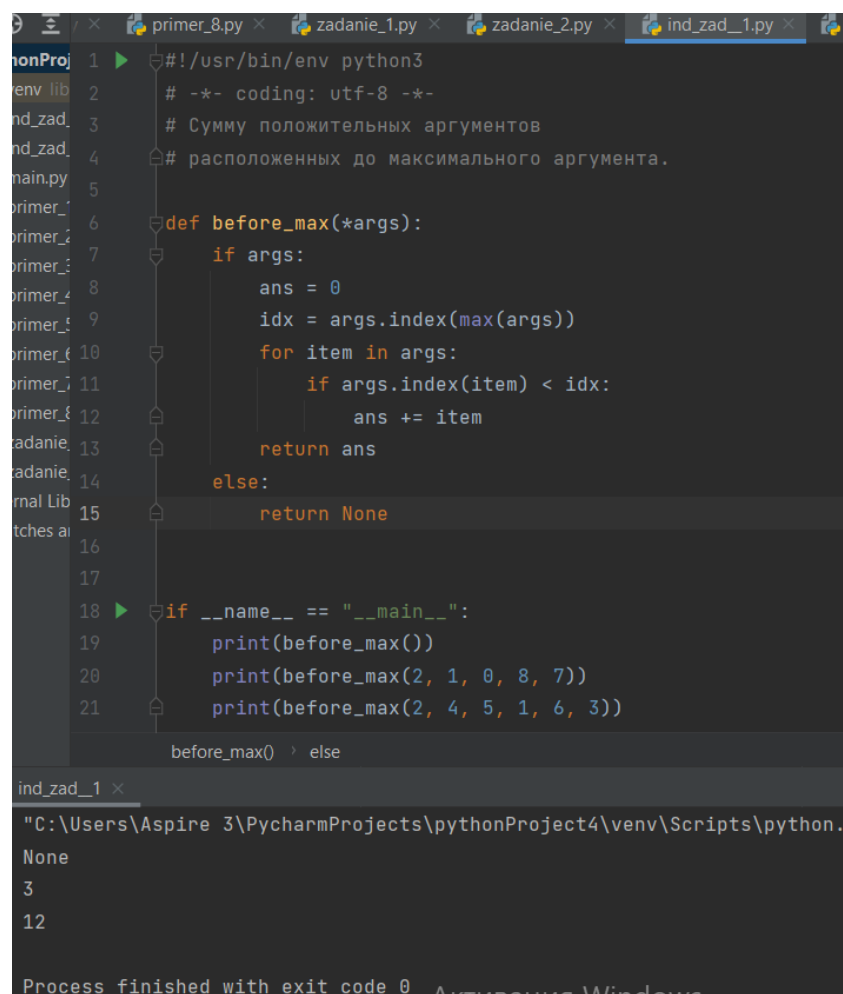
Если функции передается пустой список аргументов, то она должна возвращать значение `None`.



```
zadanie_2 x
"C:\Users\Aspire 3\PycharmProjects\pythonP
None
2.3796033994334276
None
Process finished with exit code 0
```

Рисунок 11 – Результат задачи №2

6. Решила индивидуальное задание №1. Вариант 18. Напишите функцию, принимающую произвольное количество аргументов, и возвращающую требуемое значение. Если функции передается пустой список аргументов, то она должна возвращать значение None. Сумму положительных аргументов, расположенных до максимального аргумента.

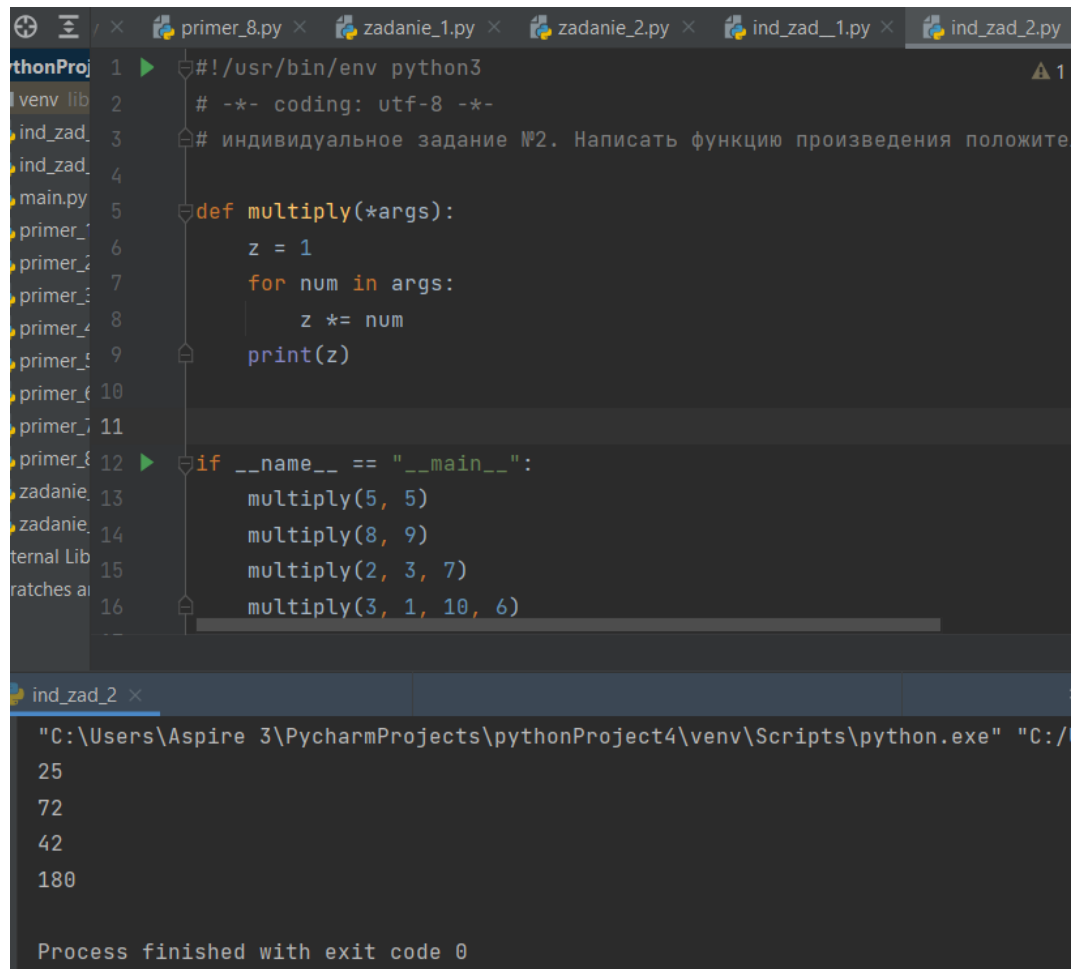


```
prim_8.py x zadanie_1.py x zadanie_2.py x ind_zad_1.py x
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 # Сумму положительных аргументов
4 # расположенных до максимального аргумента.
5
6 def before_max(*args):
7     if args:
8         ans = 0
9         idx = args.index(max(args))
10        for item in args:
11            if args.index(item) < idx:
12                ans += item
13        return ans
14    else:
15        return None
16
17
18 if __name__ == "__main__":
19     print(before_max())
20     print(before_max(2, 1, 0, 8, 7))
21     print(before_max(2, 4, 5, 1, 6, 3))
22
23 before_max() > else
```

```
ind_zad_1 x
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.
None
3
12
Process finished with exit code 0
```

Рисунок 12 – Результат и код индивидуальной задачи №1

## Индивидуально задание №2.



The screenshot shows a PyCharm IDE with several tabs open: primer\_8.py, zadanie\_1.py, zadanie\_2.py, ind\_zad\_1.py, and ind\_zad\_2.py. The active tab is ind\_zad\_2.py, which contains the following Python code:

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  # индивидуальное задание №2. Написать функцию произведения положительных
4
5  def multiply(*args):
6      z = 1
7      for num in args:
8          z *= num
9      print(z)
10
11
12 if __name__ == "__main__":
13     multiply(5, 5)
14     multiply(8, 9)
15     multiply(2, 3, 7)
16     multiply(3, 1, 10, 6)
```

Below the code editor, the output of the script is displayed in a terminal window:

```
"C:\Users\Aspire 3\PycharmProjects\pythonProject4\venv\Scripts\python.exe" "C:/
25
72
42
180

Process finished with exit code 0
```

Рисунок 13 – Результат и код индивидуальной задачи №2

Ответы на контрольные вопросы:

1. Какие аргументы называются позиционными?

Ответ: это аргументы, которые при вызове функции надо передать в правильном порядке.

2. Какие аргументы называются именованными?

Ответ: если порядок передачи аргументов по каким-то причинам не известен, то можно использовать именованные аргументы. Именованный аргумент представляет собой пару «имя-значение»

3. Для чего используется оператор \*?

Ответ: данный оператор позволяет «распаковывать» объекты, внутри которых хранятся некие элементы.

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

Ответ: каждая из этих конструкций используется для распаковки аргументов соответствующего типа, позволяя вызывать функции со списком аргументов переменной длины.

Вывод по работе: в ходе лабораторной работы были приобретены навыки по работе с функциями с переменным числом параметров при написании программ с помощью языка программирования Python. Также, были изучены именованные и позиционные аргументы.