

**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

Выполнил студент группы
ИТС-б-о-20-1 (1)

Боржонов Ростислав « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил к.т.н., доцент

Кафедры инфокоммуникаций

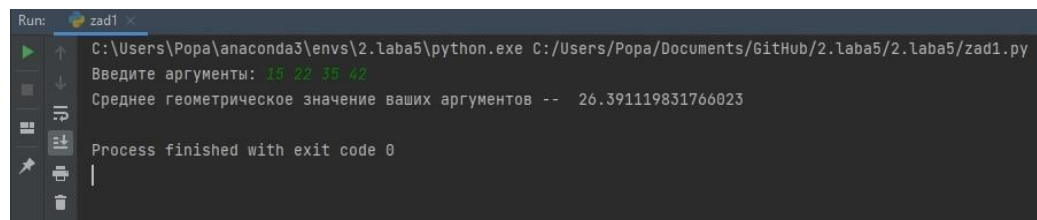
Воронкин Р.А.

(подпись)

Цель работы: приобретение навыков по работе с функциями с переменным числом параметров при написании программ с помощью языка программирования Python версии 3.x.

Ход работы:

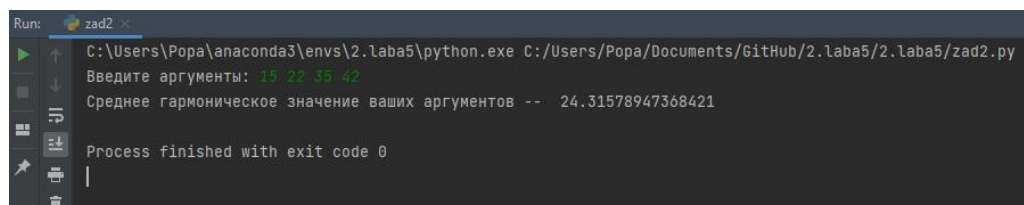
Задача 1. Решить поставленную задачу: написать функцию, вычисляющую среднее геометрическое своих аргументов.



```
Run: zad1
C:\Users\Popa\anaconda3\envs\2.laba5\python.exe C:/Users/Popa/Documents/GitHub/2.laba5/2.laba5/zad1.py
Введите аргументы: 15 22 35 42
Среднее геометрическое значение ваших аргументов -- 26.391119831766023
Process finished with exit code 0
```

Рисунок 1. Результат выполнения задачи.

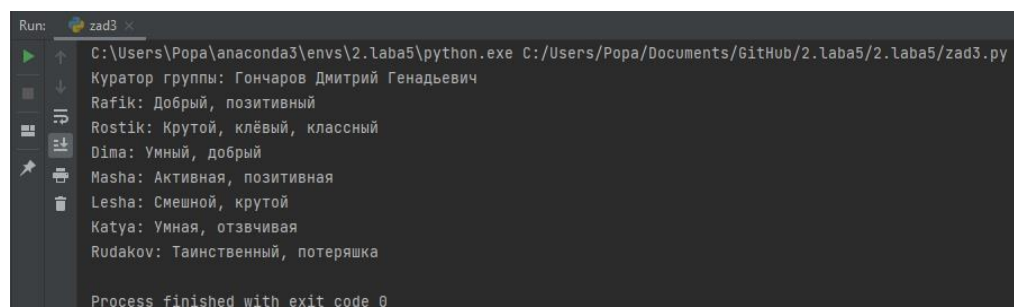
Задача 2. Решить поставленную задачу: написать функцию, вычисляющую среднее гармоническое своих аргументов.



```
Run: zad2
C:\Users\Popa\anaconda3\envs\2.laba5\python.exe C:/Users/Popa/Documents/GitHub/2.laba5/2.laba5/zad2.py
Введите аргументы: 15 22 35 42
Среднее гармоническое значение ваших аргументов -- 24.31578947368421
Process finished with exit code 0
```

Рисунок 2. Результат выполнения задачи.

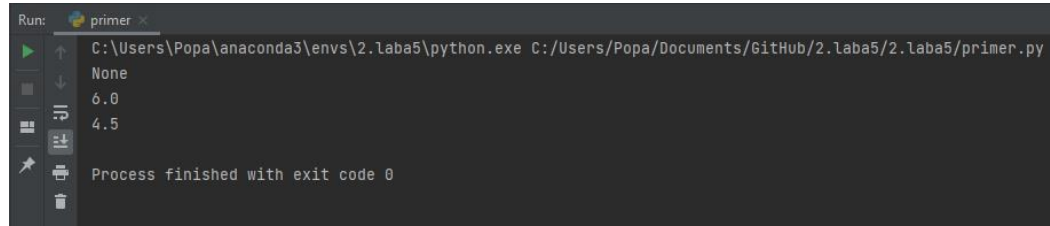
Задача 3. С помощью ****kwargs** вывести Куратора и список группы с маленькой характеристикой.



```
Run: zad3
C:\Users\Popa\anaconda3\envs\2.laba5\python.exe C:/Users/Popa/Documents/GitHub/2.laba5/2.laba5/zad3.py
Куратор группы: Гончаров Дмитрий Геннадьевич
Rafik: Добрый, позитивный
Rostik: Крутой, клёвый, классный
Dima: Умный, добрый
Masha: Активная, позитивная
Lesha: Смешной, крутой
Katya: Умная, отзывчивая
Rudakov: Таинственный, потеряшка
Process finished with exit code 0
```

Рисунок 3. Результат выполнения задачи.

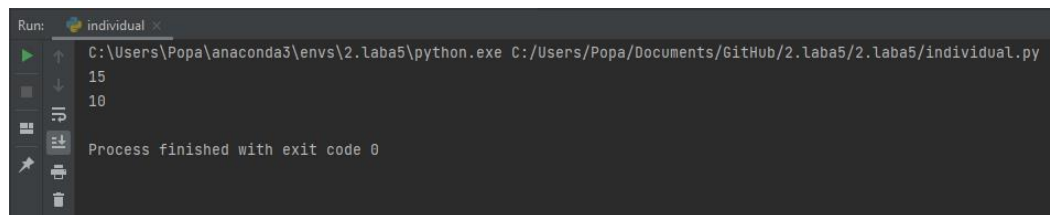
Пример.



```
Run: primer x
C:\Users\Popa\anaconda3\envs\2.laba5\python.exe C:/Users/Popa/Documents/GitHub/2.laba5/2.laba5/primer.py
None
6.0
4.5
Process finished with exit code 0
```

Рисунок 4. Результат выполнения примера.

Индивидуальное задание. Вариант 3. 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. **Сумму аргументов, расположенных между первым и последним нулевыми аргументами.**



```
Run: individual x
C:\Users\Popa\anaconda3\envs\2.laba5\python.exe C:/Users/Popa/Documents/GitHub/2.laba5/2.laba5/individual.py
15
10
Process finished with exit code 0
```

Рисунок 5. Результат выполнения индивидуального задания.

Ответы на контрольные вопросы:

1. Какие аргументы называются позиционными в Python?

Это аргументы, передаваемые в определённой последовательности (на определённых позициях), без указания их имён. Используется сокращённое название — `args` (arguments).

2. Какие аргументы называются именованными в Python?

Это аргументы, передаваемые при помощи имени (идентификатора), либо словаря с его распаковкой при помощи `**`. Используется сокращённое название — `kwargs` (keyword arguments).

3. Для чего используется оператор `*` ?

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

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

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