

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:

Логвинов Иван Васильевич

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.

Задание №1:

вычисляющую
аргументов

a_1, a_2, \dots, a_n

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

написать функцию,
среднее геометрическое своих

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

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

import math

def geometric_mean(*a):
    if a:
        nums = [num for num in a]
        out = math.pow(math.prod(nums), 1 / len(nums))
        return out
    else:
        return "None"

if __name__ == "__main__":
    print(f'Geometric mean of entered args: {geometric_mean(2, 5, 6, 8)}')
```

C:\Users\Иван\AppData\Local\Programs\Python\Python39\pythonw.exe C
Geometric mean of entered args: 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 .

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

import math

def average_harm(*a):
    if a:
        nums = [1/num for num in a]
        out = len(nums)/math.fsum(nums)
        return out
    else:
        return None

if __name__ == "__main__":
    print(f'Average of entered args: {average_harm(5, 2, 3)}')
```

C:\Users\Иван\AppData\Local\Programs\Python\Python39\py
Average of entered args: 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.

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

import math

def multiply_new(*a):
    if a:
        min_i = a.index(min(a)) + 1
        max_i = a.index(max(a))
        return math.prod(a[min_i:max_i])
    else:
        return None

if __name__ == "__main__":
    print(f'Multiply between min and max: {multiply_new(3,6,7,9)}')
```

```
C:\Users\Иван\AppData\Local\Programs\
Multiply between min and max: 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  
"""
```