

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

«СЕВЕРО-КАВКАЗСКИЙ-ФЕДЕРАЛЬНЫЙ

УНИВЕРСИТЕТ» Кафедра инфокоммуникаций

Отчет по лабораторной работе №2.10

Функции с переменным числом параметров в Python.

по дисциплине «Технологии программирования и алгоритмизация»

Выполнил студент группы ИВТ-б-о-21-1

Лысенко И.А. « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____

(подпись)

Ставрополь 2022

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

Указания к работе:

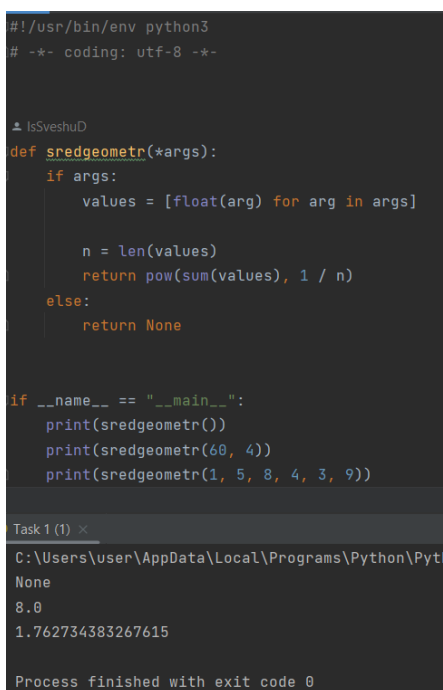
1. Решить поставленную задачу: написать функцию, вычисляющую среднее геометрическое своих аргументов. Если функции передаётся пустой список аргументов, то она должна возвращать значение None.

2. Решить поставленную задачу: написать функцию, вычисляющую среднее геометрическое своих аргументов. Если функции передаётся пустой список аргументов, то она должна возвращать значение None.

3. 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: сумму модулей аргументов, расположенных после первого аргумента, равного нулю.

Ход работы:

1. Решил первую задачу и получил следующий результат.



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

lsSveshuD
def sredgeometr(*args):
    if args:
        values = [float(arg) for arg in args]

        n = len(values)
        return pow(sum(values), 1 / n)
    else:
        return None

if __name__ == "__main__":
    print(sredgeometr())
    print(sredgeometr(60, 4))
    print(sredgeometr(1, 5, 8, 4, 3, 9))

Task 1 (1) x
C:\Users\user\AppData\Local\Programs\Python\Python38\python.exe
None
8.0
1.762734383267615

Process finished with exit code 0
```

Рисунок 1 – Среднее геометрическое

2. Решил вторую задачу и получил следующий результат.

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

import numpy as np

def garmony(*args):
    if args:
        values = [float(arg) for arg in args]

        n = len(values)
        p = np.prod(values)
        return n / (sum(values) / p)
    else:
        return None

if __name__ == "__main__":
    print(garmony())
    print(garmony(60, 4))
    print(garmony(1, 5, 8, 4, 3, 9))
```

Task 2 (1) x

C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe

None
7.5
864.0

Рисунок 2 – Среднее гармоническое

3. Нашёл сумму аргументов, расположенных после минимального аргумента.

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

def sm(*args):
    if args:
        values = [float(arg) for arg in args]

        print(sum(values[values.index(min(values)) + 1:]))
    else:
        return None

if __name__ == "__main__":
    print(sm())
    print(sm(3, 0, -6, 9))
    print(sm(1, 0, 0, 4, 3, 9))
```

Individual Task 2021 x

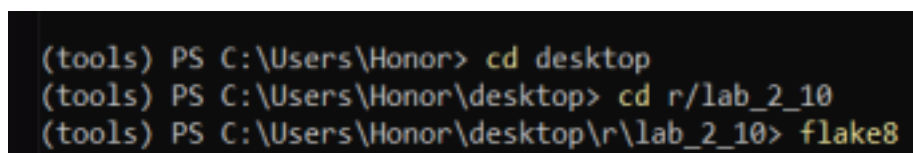
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe

None
9.0
None
16.0
None

Process finished with exit code 0

Рисунок 3 – Результат выполнения индивидуального задания

4. Проверил с помощью flake8.



```
(tools) PS C:\Users\Honor> cd desktop  
(tools) PS C:\Users\Honor\desktop> cd r\lab_2_10  
(tools) PS C:\Users\Honor\desktop\r\lab_2_10> flake8
```

Рисунок 4 – Проверка flake8.

Ответы на вопросы

1. Какие аргументы называются позиционными в Python? При вызове функции в Python, каждому аргументу должен быть поставлен в соответствие параметр в определении функции. Проще всего это можно сделать на основании порядка перечисления аргументов. Значения, связываемые с аргументами подобным образом, называют — позиционными аргументами.

2. Какие аргументы называются именованными в Python? Именованный аргумент представляет собой пару «имя-значение», передаваемую функции. Имя и значения связываются с аргументом напрямую, так что при передаче аргумента путаница с порядком исключается.

3. Для чего используется оператор *?

Он позволяет распаковывать объекты, внутри которых хранятся некие элементы.

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

*args — это сокращение от «arguments» (аргументы), а **kwargs — сокращение от «keyword arguments» (именованные аргументы). Каждая из этих конструкций используется для распаковки аргументов соответствующего типа, позволяя вызывать функции со списком аргументов переменной длины.

Вывод: приобрёл навыки по работе с функциями с переменным числом параметров при написании программ с помощью языка программирования Python.