

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

Кафедра инфокоммуникаций

Отчет по лабораторной работе №5

Функции с переменным числом параметров в языке Python

По дисциплине «Технологии программирования и алгоритмизация»

Выполнил студент группы ИВТ-б-о-20-1

Погорелов Д.Н « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р. А. _____

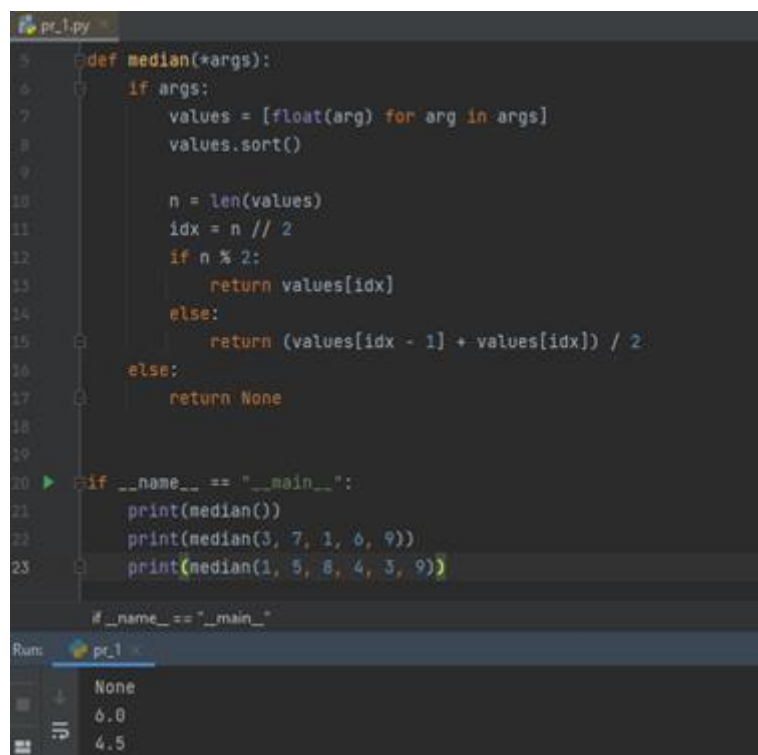
(подпись)

Ставрополь 2021

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:

1. Изучил теоретический материал для выполнения лабораторной работы.
2. Создал общедоступный репозиторий
3. Проработал пример, приведенный в теоретическом материале.



```
pr_1.py
3 def median(*args):
4     if args:
5         values = [float(arg) for arg in args]
6         values.sort()
7
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
19
20 if __name__ == "__main__":
21     print(median())
22     print(median(3, 7, 1, 6, 9))
23     print(median(1, 5, 8, 4, 3, 9))
24
25 if __name__ == "__main__":
26     print(median())
27     print(median(3, 7, 1, 6, 9))
28     print(median(1, 5, 8, 4, 3, 9))
```

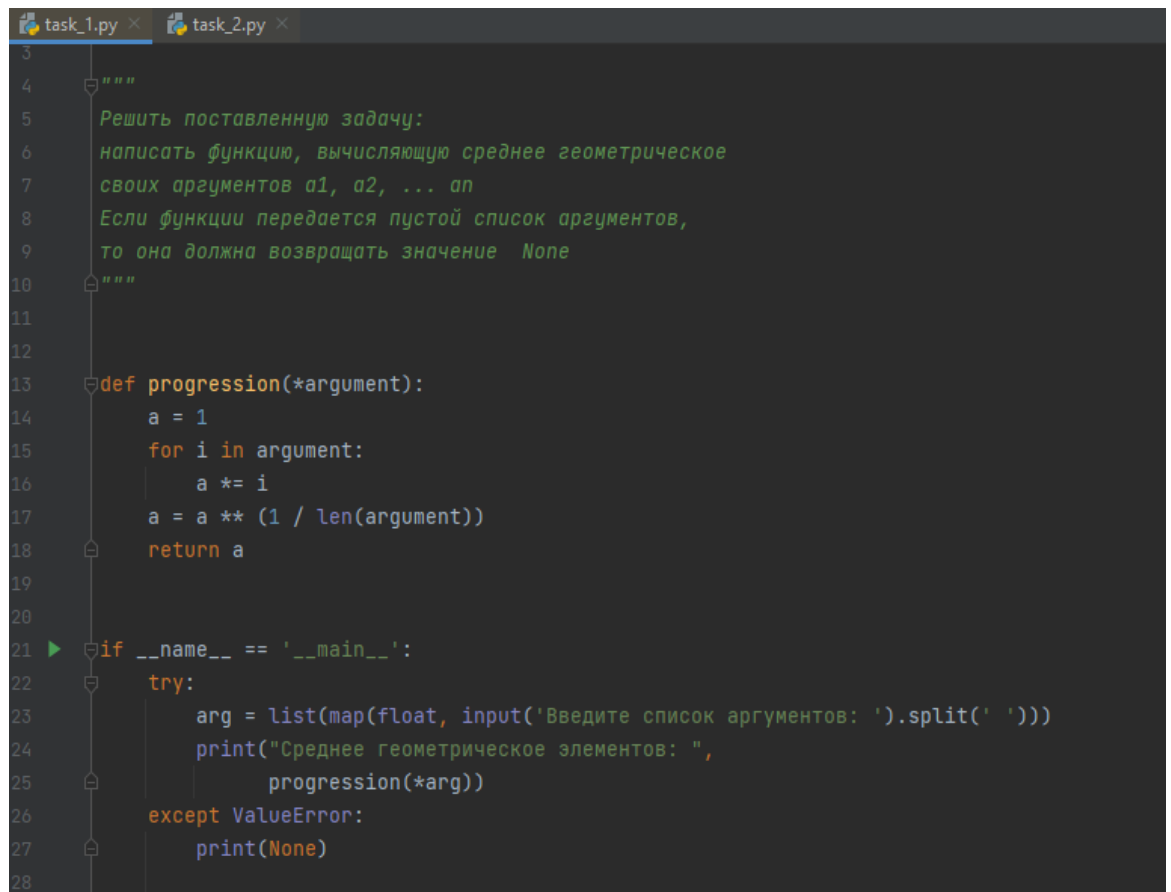
Run: pr_1

None
6.0
4.5

Рисунок 1 – Результат работы примера

4. Приступил к выполнению общих заданий.
5. Написал код для выполнения первого задания

Условие: написать функцию, вычисляющую среднее геометрическое своих аргументов a_1, a_2, \dots, a_n . Если функции передается пустой список аргументов, то она должна возвращать значение None.

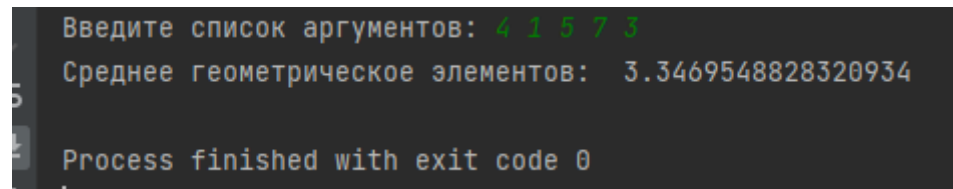


```

3
4 """
5 Решить поставленную задачу:
6 написать функцию, вычисляющую среднее геометрическое
7 своих аргументов a1, a2, ... an
8 Если функции передается пустой список аргументов,
9 то она должна возвращать значение None
10 """
11
12
13 def progression(*argument):
14     a = 1
15     for i in argument:
16         a *= i
17     a = a ** (1 / len(argument))
18     return a
19
20
21 if __name__ == '__main__':
22     try:
23         arg = list(map(float, input('Введите список аргументов: ').split(' ')))
24         print("Среднее геометрическое элементов: ",
25               progression(*arg))
26     except ValueError:
27         print(None)
28

```

Рисунок 2 – Код для выполнения задания



```

Введите список аргументов: 4 1 5 7 3
Среднее геометрическое элементов: 3.3469548828320934
Process finished with exit code 0

```

Рисунок 3 – Результат выполнения задания

6. Приступил к выполнению второго задания.

Условие: написать функцию, вычисляющую среднее гармоническое своих аргументов a_1, a_2, \dots, a_n . Если функции передается пустой список аргументов, то она должна возвращать значение `None`.

```

13 def harmonic_mean(*args):
14     x = 0
15     for i in args:
16         if i == 0:
17             return "Ошибка! В функции присутствует 0"
18         else:
19             x += 1 / float(i)
20     harmonic = 1 / (1 / len(args) * x)
21     return harmonic
22
23
24 if __name__ == '__main__':
25     try:
26         arg = list(map(float, input('Введите список аргументов: ').split(' ')))
27         print("Среднее гармоническое элементов: ",
28               harmonic_mean(*arg))
29     except ValueError:
30         print(None)
31

```

Рисунок 4 – Код для работы второго задания

```

Введите список аргументов: 4 9 11 4
Среднее гармоническое элементов: 5.697841726618705

Process finished with exit code 0

```

Рисунок 5 – Результат работы кода для второго задания

7. Выполнил третье задание. Условие: найти среднее арифметическое чисел, а также их сумму.

```

1  ▶  #!/usr/bin/env python3
2      # -*- coding: utf-8 -*-
3
4      """
5      Найти среднее арифметическое чисел.
6      """
7
8
9  def arithmetic(**kwargs):
10     x = 0
11     k = 0
12     len_of = len(kwargs)
13     for numbers in kwargs.values():
14         x += float(numbers)
15     sum_of = x / len_of
16
17     for n in kwargs.values():
18         k += float(n)
19     tip = k
20     print(f"Сумма чисел: {tip}")
21     print('Среднее арифметическое чисел: ', sum_of)
22
23
24  ▶  if __name__ == '__main__':
25     arithmetic(a=5, b=10, c=4, d=1)

```

Рисунок 6 – Код для решения задачи

```

Сумма чисел: 20.0
Среднее арифметическое чисел:  5.0

Process finished with exit code 0

```

Рисунок 7 – Результат решения задачи

Индивидуальное задание. Вариант 1.

Условие: напишите функцию, принимающую произвольное количество аргументов, и возвращающую произведение аргументов, расположенных между максимальным и минимальным аргументами. 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.

1. Ознакомившись с условиями задачи, приступил к выполнению задания.

2. Написал код для решения задачи.

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

def min_max(*args):
    if not args:
        return None

    multi = 1

    maximum = args[0]
    max_ind = 0

    minimum = args[0]
    min_ind = 0

    for i, item in enumerate(args):
        if item > maximum:
            maximum = item
            max_ind = i
        if item < minimum:
            minimum = item
            min_ind = i
    for i in args[min_ind:max_ind]:
        multi *= i
    return multi

if __name__ == '__main__':
    print("Введите список аргументов: ")
    arg = list(map(float, input().split()))
    print(min_max(*arg))
```

Рисунок 6 – Код для выполнения задания

```
Введите список аргументов: 4 3 2 1 2 8 9 11 3
Произведение аргументов, расположенных между максимальным и минимальным аргументами: 144.0
Process finished with exit code 0
```

Рисунок 7 – Результат выполнения задачи

Контрольные вопросы:

1. Какие аргументы называются позиционными в Python?

Позиционные аргументы обрабатываются слева направо. То есть оказывается, что позиция аргумента, переданного функции, находится в прямом соответствии с позицией параметра, использованного в заголовке функции при её объявлении.

2. Какие аргументы называются именованными в Python?

Именованные аргументы передают функциям с указанием имён этих аргументов, соответствующих тем именам, которые им назначены при объявлении функции.

3. Для чего используется оператор *?

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

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

При применении конструкции *args в параметр args попадают позиционные аргументы, представляемые в виде кортежа. При применении **kwargs в kwargs попадают именованные аргументы, представленные в виде словаря.

Вывод: в ходе выполнения лабораторной работы приобретены навыки по работе с функциями и с переменным числом параметров при написании программ с помощью языка программирования Python версии 3.x.