

**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
по дисциплине «Основы программной инженерии»**

Выполнил студент группы
ПИЖ-б-о-21-1

Трушева В. О. .« » 2022г.

Подпись студента _____

Работа защищена «
» _____ 20__ г.

Проверила Воронкин Р.А.

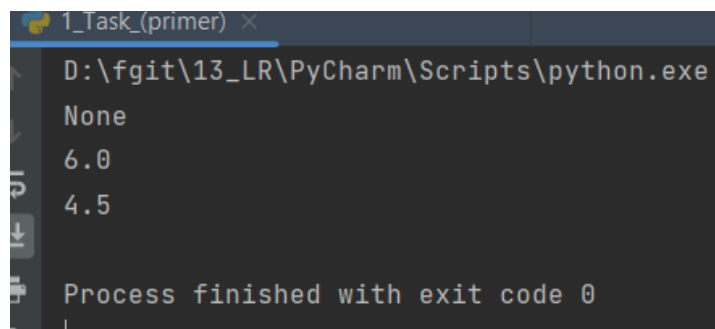
(подпись)

Ставрополь 2022

Методика и порядок выполнения работы

1. Изучить теоретический материал работы.
2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.
3. Выполните клонирование созданного репозитория.
4. Дополните файл .gitignore необходимыми правилами для работы с IDE PyCharm.
5. Организуйте свой репозиторий в соответствие с моделью ветвления git-flow.
6. Создайте проект PyCharm в папке репозитория.
7. Проработать примеры лабораторной работы.

Условие. Разработать функцию для определения медианы значений аргументов функции. Если функции передается пустой список аргументов, то она должна возвращать значение None.



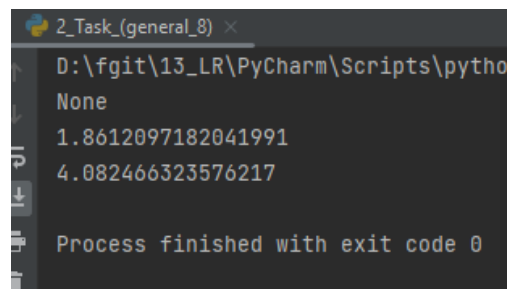
```
1_Task_(primer) x
D:\fgit\13_LR\PyCharm\Scripts\python.exe
None
6.0
4.5
Process finished with exit code 0
```

Рисунок 1 – Результат выполнения программы

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

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

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



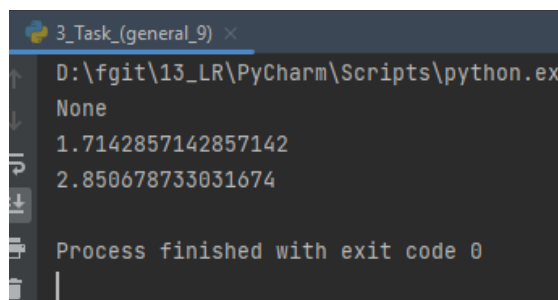
```
2_Task_(general_8) x
D:\fgit\13_LR\PyCharm\Scripts\python
None
1.8612097182041991
4.082466323576217
Process finished with exit code 0
```

Рисунок 2 – Результат выполнения программы

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

$$\frac{n}{H} = \sum_{k=1}^n \frac{1}{a_k}.$$

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



```
3_Task_(general_9) x
D:\fgit\13_LR\PyCharm\Scripts\python.exe
None
1.7142857142857142
2.850678733031674
Process finished with exit code 0
|
```

Рисунок 3 – Результат выполнения программы

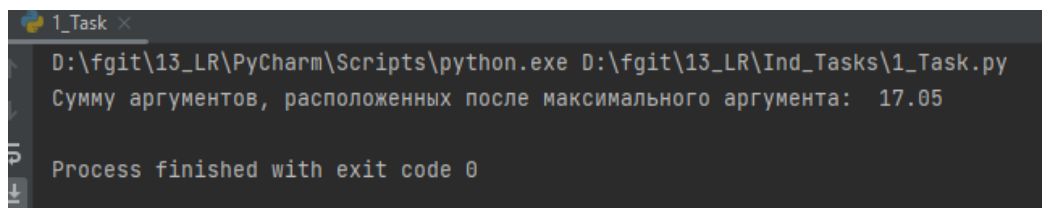
10. Приведите в отчете скриншоты результатов выполнения примера при различных исходных данных вводимых с клавиатуры.

11. Зафиксируйте изменения в репозитории.

12. Решите индивидуальное задание согласно своего варианта.

Вариант – 12

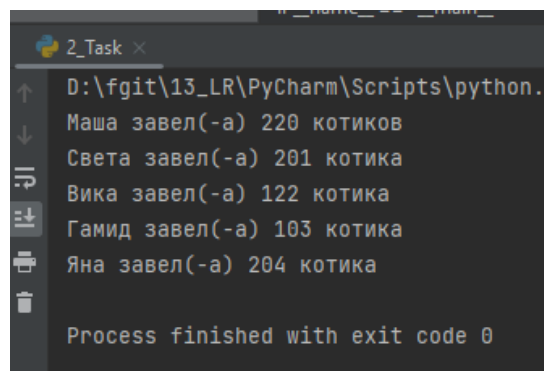
Условие. Сумму аргументов, расположенных после максимального аргумента.



```
1_Task x
D:\fgit\13_LR\PyCharm\Scripts\python.exe D:\fgit\13_LR\Ind_Tasks\1_Task.py
Сумму аргументов, расположенных после максимального аргумента: 17.05
Process finished with exit code 0
```

Рисунок 4 – Результат выполнения программы

13. Самостоятельно подберите или придумайте задачу с переменным числом именованных аргументов. Приведите решение этой задачи.



```
2_Task x
D:\fgit\13_LR\PyCharm\Scripts\python.exe
Маша завел(-а) 220 котиков
Света завел(-а) 201 котика
Вика завел(-а) 122 котика
Гамид завел(-а) 103 котика
Яна завел(-а) 204 котика
Process finished with exit code 0
```

Рисунок 5 – Результат выполнения программы

14. Зафиксируйте изменения в репозитории.

15. Добавьте отчет по лабораторной работе в формате PDF в папку doc репозитория. Зафиксируйте изменения.

16. Выполните слияние ветки для разработки с веткой master/main.

17. Отправьте сделанные изменения на сервер GitHub.

18. Отправьте адрес репозитория GitHub на электронный адрес преподавателя.

Контрольные вопросы:

1. Какие аргументы называются позиционными в Python?

Это аргументы, передаваемые в вызов в определённой последовательности (на определённых позициях), без указания их имён. Элементы объектов, поддерживающих итерирование, могут использоваться в качестве позиционных аргументов, если их распаковать при помощи `*`.

2. Какие аргументы называются именованными в Python?

Это аргументы, передаваемые в вызов при помощи имени (идентификатора), либо словаря с его распаковкой при помощи `**`.

3. Для чего используется оператор `*`?

Функция также может принимать переменное количество позиционных аргументов, тогда перед именем ставится `*`.

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.