

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

Институт цифрового развития

ОТЧЁТ

по лабораторной работе №2.10

Дисциплина: «Программирование на Python»

Тема: «»

Выполнил: студент 2 курса

группы ИВТ-б-о-21-1

Толубаев Рамиль Ахметович

Ставрополь 2022

1. Создал репозиторий с использованием лицензии MIT. Затем клонировал его на ПК:

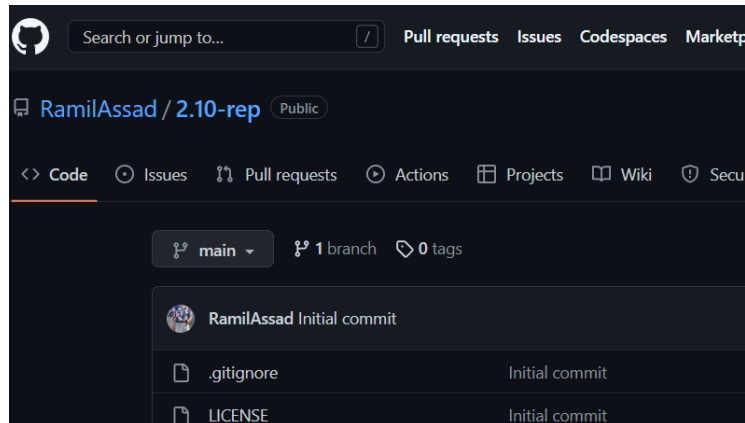


Рисунок 1. Новый репозиторий

2. Решение 1 задачи

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

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

Тест функции работает с такой список аргументов, то же самое работает с любым N

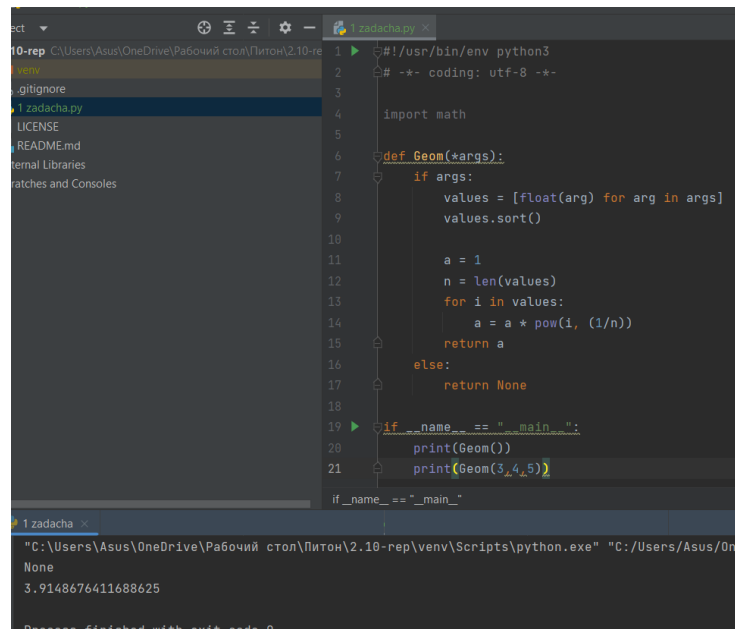
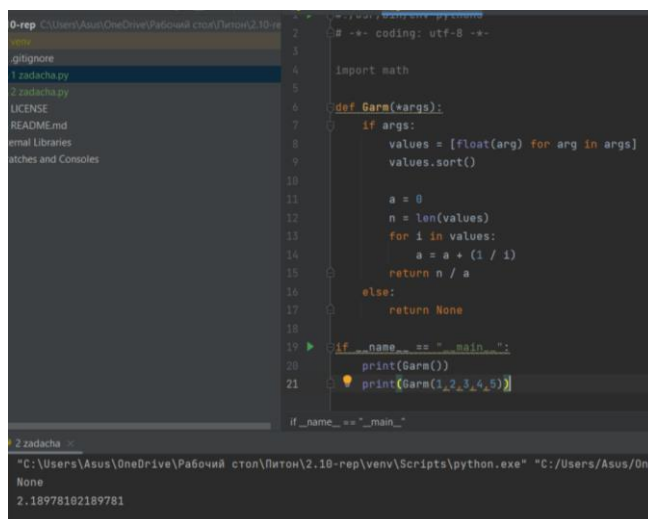


Рисунок 2. Решение 1 задачи

3. Решение 2 задачи

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

$$\frac{n}{H} = \sum_{k=1}^n \frac{1}{a_k}. \quad (2)$$



```
1 # -*- coding: utf-8 -*-
2
3 import math
4
5 def Garm(*args):
6     if args:
7         values = [float(arg) for arg in args]
8         values.sort()
9
10        a = 0
11        n = len(values)
12        for i in values:
13            a = a + (1 / i)
14        return n / a
15    else:
16        return None
17
18 if __name__ == "__main__":
19     print(Garm())
20     print(Garm(1, 2, 3, 4, 5))
21
22 if __name__ == "__main__":
```

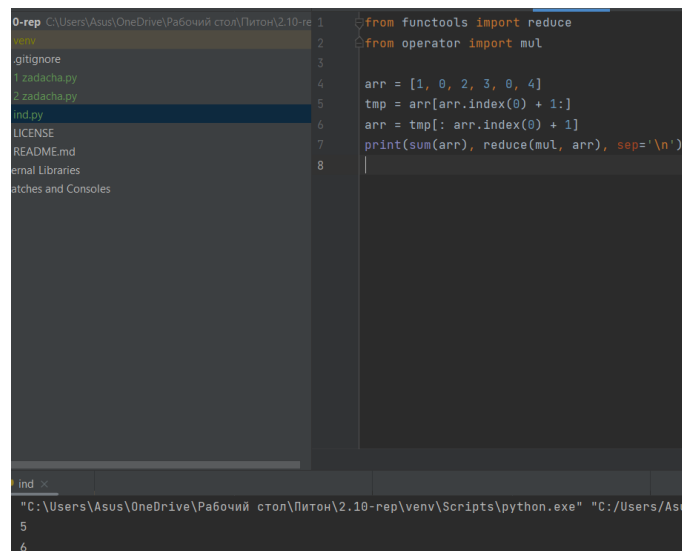
Рисунок 3. Решение 2 задачи

4. Индивидуальное задание

Вариант 20 (2)

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.

2. Сумму аргументов, расположенных между первым и последним нулевыми аргументами.

A screenshot of a code editor with a dark theme. The left sidebar shows a file explorer with a tree view containing files like .gitignore, 1.zadacha.py, 2.zadacha.py, ind.py, LICENSE, README.md, External Libraries, and Snippets and Consoles. The main editor area shows a Python script with the following code:

```
1 from functools import reduce
2 from operator import mul
3
4 arr = [1, 0, 2, 3, 0, 4]
5 tmp = arr[arr.index(0) + 1:]
6 arr = tmp[: arr.index(0) + 1]
7 print(sum(arr), reduce(mul, arr), sep='\n')
8
```

The bottom of the editor shows a terminal window with the command `python.exe "C:/Users/Asus/OneDrive/Рабочий стол/Питон/2.10-rep/venv/Scripts/python.exe" "C:/Users/Asus/OneDrive/Рабочий стол/Питон/2.10-rep/venv/Scripts/python.exe"` and the output:

```
5
6
```

Рисунок 4. Решение индивидуальной задачи

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.