

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ №10
дисциплины «Анализ данных»

Выполнил:
Степанов Леонид Викторович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение
средств вычислительной
техники и автоматизирование
систем», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р.А., канд. техн. наук,
доцент, доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2024 г.

Тема: Синхронизация потоков в языке программирования Python

Цель работы: приобретение навыков использования примитивов синхронизации в языке программирования Python версии 3.x.

Ход работы:

Индивидуальное задание: для своего индивидуального задания лабораторной работы 2.23 необходимо организовать конвейер, в котором сначала в отдельном потоке вычисляется значение первой функции, после чего результаты вычисления должны передаваться второй функции, вычисляемой в отдельном потоке. Потоки для вычисления значений двух функций должны запускаться одновременно.

Создал файл (ind.py) в котором при помощи barrier организовал конвейер, в котором вычисляется сначала натуральный логарифм от 3 в степени номера текущего члена ряда, далее значение передается в функцию возведения в степень после чего в функцию нахождение факториала.

На рис. 1 представлена функция реализующая конвейер:

```
15 br = Barrier(4)
16
17
18 def calculate_sum(x):
19     return 3**x
20
21
22 def calculate_part(results, x, cur):
23     local_result = [1]
24
25     def my_log(local_result):
26         local_result[0] *= math.pow(math.log(3), cur)
27         br.wait()
28
29     def my_pow(local_result):
30         local_result[0] *= x**cur
31         br.wait()
32
33     def my_fact(local_result):
34         local_result[0] /= math.factorial(cur)
35         br.wait()
36
37     Thread(target=my_log, args=(local_result,)).start()
38     Thread(target=my_pow, args=(local_result,)).start()
39     Thread(target=my_fact, args=(local_result,)).start()
40
41     br.wait()
42     results.append(local_result[0])
43
```

Рисунок 1 – Функция calculate_part

Вывод: в ходе выполнения практической работы были приобретены навыки использования примитивов синхронизации в языке программирования Python версии 3.x.