

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)**

Кафедра автоматизированных систем управления (АСУ)

Отчёт по лабораторной работе №1

на тему «Решение уравнений методом трапеции»

по дисциплине «Разработка веб-сервисов для научных и прикладных задач»

Студент гр. 438-3

Гурачевский Д.С.

25 марта 2020 г.

Руководитель

Преподаватель каф. АСУ

Суханов А.Я.

25 марта 2020 г.

Томск 2020

1 Цели и задачи работы

1.1 Тема

Лабораторная работа 1

1.2 Цель

Первое знакомство с языком Python и освоение простых возможностей языка.

Краткое знакомство с соглашениями о написании кода на Python PEP8.

Краткое знакомство с онлайн интерпретаторами.

Вывод и ввод информации.

Основные операции.

Первое знакомство с массивами numpy на примере интегрирования функций.

Знакомство с возможностями построения графиков в matplotlib.

1.3 Задача

Написать программу, которая считает интеграл без использования массивов numpy.

Оценить время исполнения с использованием numpy и без использования, для сравнения рассчитать интегралы по нескольким функциям в массивах numpy.

Реализовать сами функции. Для оценки времени исполнения воспользоваться модулем time.

2 Код программы

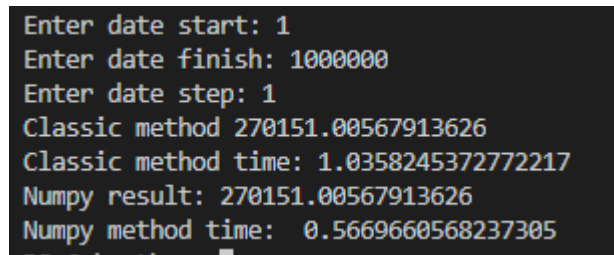
```
import time
import numpy as np
import math

def classic_method(a, b, step, func):
    start = time.time()
    if func == 'sin':
        print('Classic method',sum(math.sin(i)*math.sin(i+step) for i in range(a, b
-step, step))/step)
    elif func == 'cos':
        print('Classic method',sum(math.cos(i)*math.cos(i+step) for i in range(a, b
-step, step))/step)
    else:
        raise('Error')
    finish = time.time() - start
    return finish

def numpy_method(a, b, step, func):
    if func == 'sin':
        start = time.time()
        y = np.array([i for i in range(a, b, step)])
        y = np.sin(y)
        result = sum(y[: -1]*y[1:])/step
        print('Numpy result:',result)
    elif func == 'cos':
        start = time.time()
        y = np.array([i for i in range(a, b, step)])
        y = np.cos(y)
        result = sum(y[: -1]*y[1:])/step
        print('Numpy result:',result)
    else:
        raise('Error')
    finish = time.time() - start
    return finish

print('Enter date start:', end=' ')
a = int(input())
print('Enter date finish:', end=' ')
b = int(input())
print('Enter date step:', end=' ')
step = int(input())
print('Classic method time:',classic_method(a, b, step, 'sin'))
print('Numpy method time: ',numpy_method(a, b, step, 'sin'))
```

3 Скрин программы



```
Enter date start: 1
Enter date finish: 1000000
Enter date step: 1
Classic method 270151.00567913626
Classic method time: 1.0358245372772217
Numpy result: 270151.00567913626
Numpy method time: 0.5669660568237305
```

Рисунок 1 - Вывод результата работы программы

4.Вывод

Познакомился с языком Python и освоил простых возможностей языка.

Кратко ознакомился с соглашениями о написании кода на Python PEP8.

Познакомился с библиотекой numpy на примере интегрирования функций.