МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Информатика»

**Отчет по лабораторной работе №2.3**

**по дисциплине**

**«Базовые средства математических пакетов»**

Выполнил: студент гр. БЭИ2202

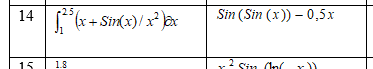
Кулешов А. С.

Вариант 14.

Проверил: доцент Шакин В. Н.

Москва, 2024г.

1. **Цель работы:** Изучение способов создания полиномов в Scilab, операций и функций над полиномами, нахождения корней, производных от полиномов.
2. **Индивидуальное задание**
3. ***Изучить материал учебника*** [1](*п. 2.3).*
4. ***Выбрать*** *из* ***табл. 2.3-1*** *вариант индивидуального задания.*
5. ***Создать сценарии*** *для выполнения практического задания.*
6. ***Описать*** *подынтегральную функцию***f(x)***.*
7. ***Вычислить*** *определенный интеграл с использованием функции* **inttrap(x,y)***, реализующей формулу трапеций, получив предварительно таблицу значений подынтегральной функции с шагом* **h***, и задав значения аргумента и функции в виде векторов.*
8. ***Вычислить*** *значение определенного интеграла с различной точностью, используя функцию* **integrate('f',a,b)***, реализующей формулу Симпсона, где* **f** *– имя подынтегральной функции, взятое в одинарные кавычки.*
9. ***Вычислить*** *производную от функции* **y(x)***в нескольких точках, задавая их значения как вектор аргументов. Затем вычислить производную в этих же точках по отдельности. Сравнить результаты.*
10. ***Спроектировать и реализовать приложение****:* ***«Интегрирование и дифференцирование в среде Scilab»*** *для ввода данных и отображения результатов (по требованию преподавателя).*
11. ***Предоставить*** *результаты работы преподавателю и* ***ответить*** *на поставленные вопросы.*
12. ***Оформить отчет*** *по выполненной работе****.***



1. **Программный код**

import numpy as np  
from scipy.integrate import quad  
  
# Описание подынтегральной функции f(x)  
def f(x):  
 return x + np.sin(x)/x\*\*2  
  
# Вычисление определенного интеграла с использованием метода трапеций  
def inttrap(x, y):  
 return np.trapz(y, x)  
  
# Задание пределов интегрирования и шага  
a = 1  
b = 2.5  
h = 0.01  
  
# Вычисление значений подынтегральной функции на интервале [a, b]  
x\_values = np.arange(a, b + h, h)  
y\_values = f(x\_values)  
  
# Вычисление определенного интеграла методом трапеций  
integral\_trap = inttrap(x\_values, y\_values)  
print("Определенный интеграл методом трапеций:", integral\_trap)  
  
# Вычисление определенного интеграла методом Симпсона  
integral\_simpson, error = quad(f, a, b)  
print("Определенный интеграл методом Симпсона:", integral\_simpson)  
  
# Определение функции y(x)  
def y(x):  
 return np.sin(np.sin(x)) - 0.5\*x  
  
# Задание точек, в которых вычисляется производная  
x\_points = np.array([1, 1.5, 2.5])  
  
# Функция для численного дифференцирования (центральная разностная схема)  
def numerical\_derivative(f, x, dx = 1e-6):  
 return (f(x + dx) - f(x - dx)) / (2 \* dx)  
  
# Вычисление производной в нескольких точках  
derivatives\_vectorized = numerical\_derivative(y, x\_points)  
  
# Вычисление производной в каждой точке по отдельности  
derivatives\_separately = [numerical\_derivative(y, x\_point) for x\_point in x\_points]  
  
# Вывод результатов  
for i in range(len(x\_points)):  
 print("Точка:", x\_points[i])  
 print("Производная (векторизованная):", derivatives\_vectorized[i])  
 print("Производная (по отдельности):", derivatives\_separately[i])

1. **Результаты выполнения программы**

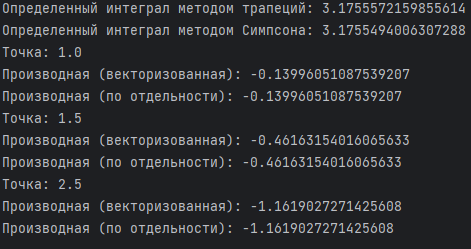
****

Рисунок 1 – Полученный результат