Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет информационных технологий Кафедра прикладной математики

Отчет защищен с оценкой

Преподаватель

(подпись)

« » 2023 г.

Отчет

По лабораторной работе №5

# «Вычисление определенных интегралов методами прямоугольников, трапеций и Симпсона»

по дисциплине «Вычислительные алгоритмы»

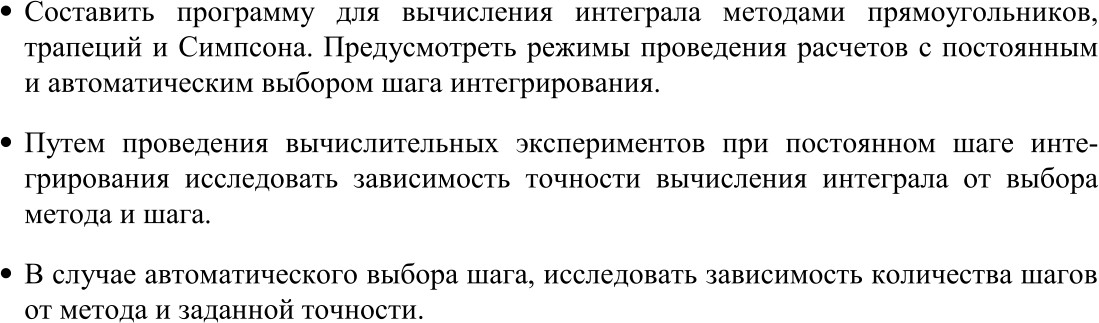
Студент группы ПИ-02 Чередов Р. А.

Преподаватель

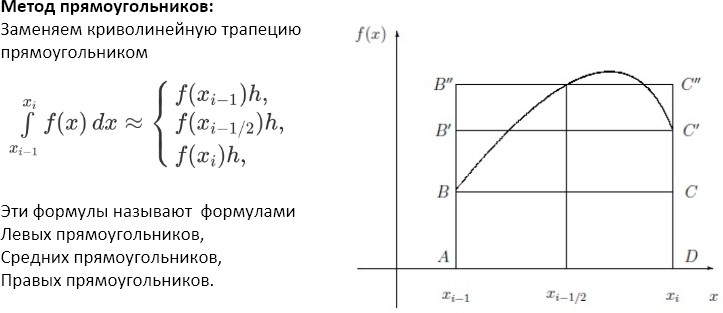
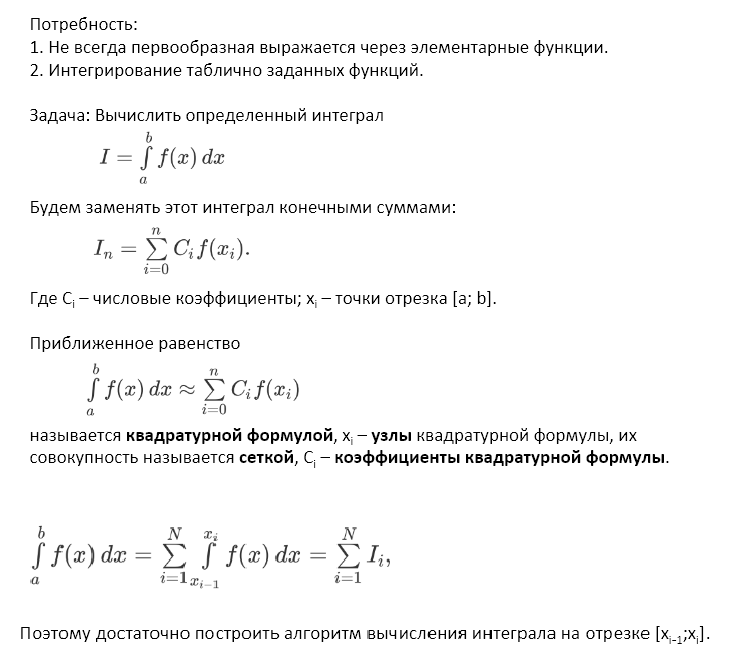
Проскурин А. В.

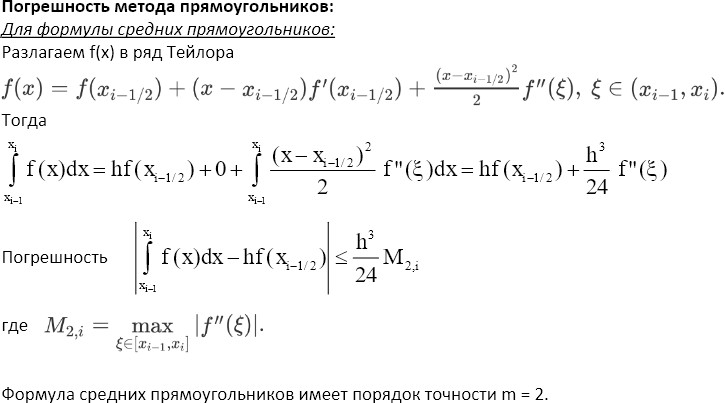
Барнаул 2023

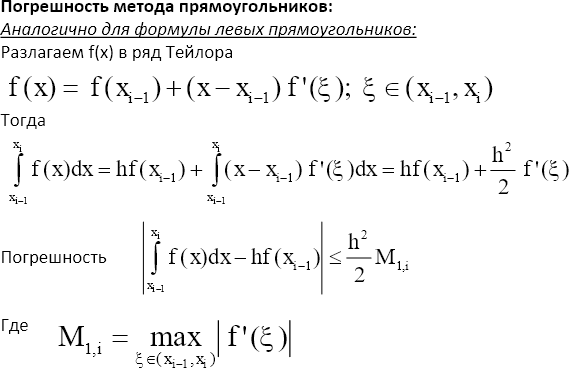
# Задание к лабораторной работе:



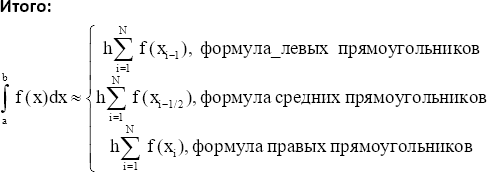
**Описание метода:**

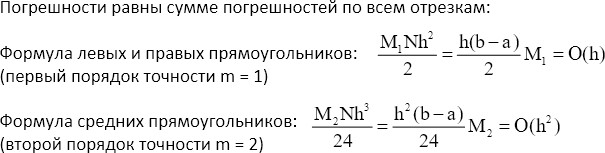


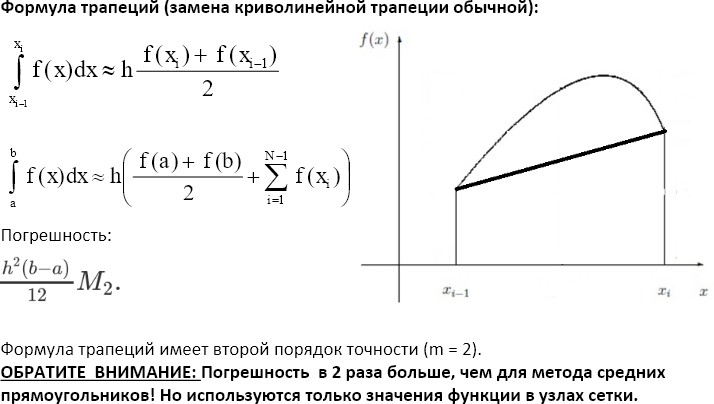


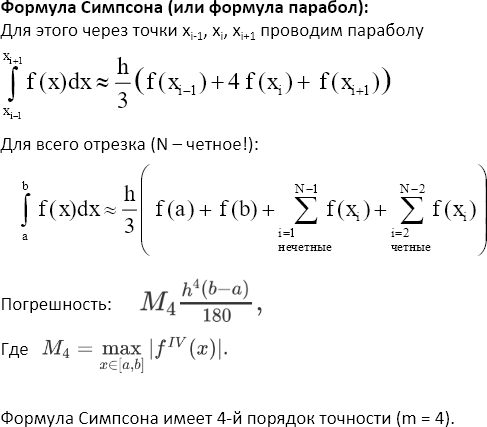












# Программа:

import matplotlib.pyplot as plt import numpy as np

from math import \*

def inpXY (a, b, step):

# Массив значений х на промежутке [a, b + step) с шагом step X = np.arange(a, b, step)

Y = []

# Массив значений функции f в точках X[] for x in X:

Y.append(eval(f)) return Y

def sum (Y): sumY = 0 for y in Y:

sumY += y return sumY

def autoStep(a, b, eps): step0 = 3

step = step0

Y = inpXY(a, b + step, step) sumY = sum(Y)

res = (sumY - Y[len(Y) - 1]) \* step while True:

step /= 5

Y = inpXY(a, b + step, step) sumY = sum(Y)

res1 = (sumY - Y[len(Y) - 1]) \* step if (abs(res1 - res)) < eps:

res = res1 break

res = res1

print("\nМетод левых прямоугольников с автошагом: ", format(res, '.3f'), "\nНачальный шаг: ", step0, "\nИтоговый шаг: ", format(step, '.3f'))

f = input("Введите функцию для интегрирования: ")

a = float(input("Введите нижний предел интегрирования: ")) b = float(input("Введите верхний предел интегрирования: ")) step = float(input("Введите шаг разбиения: "))

eps = float(input("Введите точность: "))

# Массив значений функции f в точках X[] Y = inpXY(a, b + step, step)

# Массив значений функции f в точках сX[] cY = inpXY(a + step / 2, b + step / 2, step)

# Сумма всех значений функции Y[] sumY = sum(Y)

# Сумма всех значений функции сY[] sumCY = sum(cY)

#Метод левых прямоугольников lRect = (sumY - Y[len(Y) - 1]) \* step autoStep(a, b, eps)

#Метод средних прямоугольников cRect = sumCY \* step

#Метод правых прямоугольников rRect = (sumY - Y[0]) \* step

print("\nМетод левых прямоугольников: ", str(lRect), "\nМетод средних прямоугольников: ", str(cRect), "\nМетод правых прямоугольников: ", str(rRect))

#Метод трапеций

trap = ((Y[0] + Y[len(Y) - 1]) / 2 + (sumY - (Y[0] + Y[len(Y) - 1]))) \* step print("\nМетод трапеций: ", str(trap))

# Метод Симпсона

simp = Y[0] + Y[len(Y) - 1] s1 = 0

s2 = 0

# Нечетные

for i in range(1, int((len(Y)) / 2)): s1 += Y[2 \* i - 1]

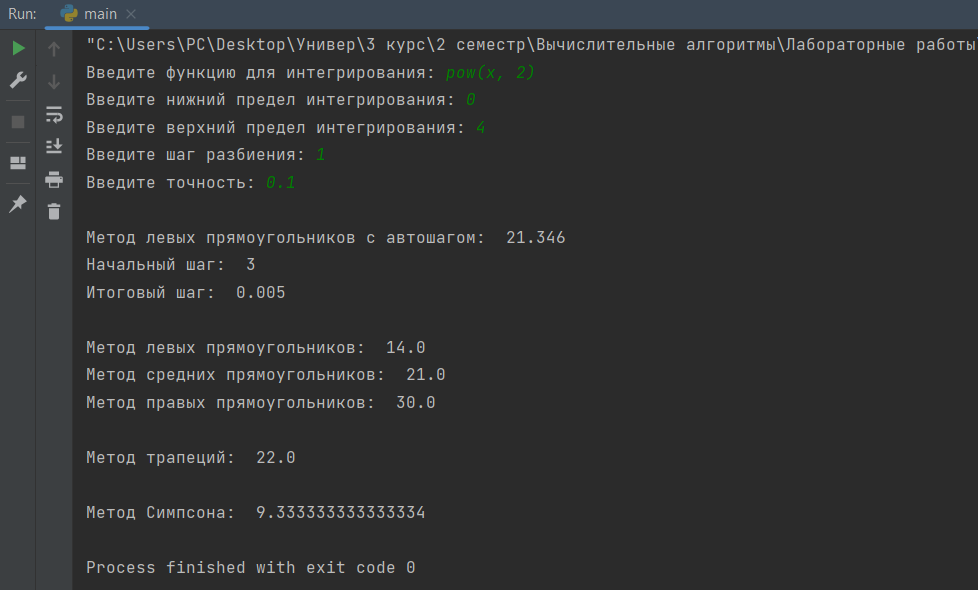
# Четные

for i in range(1, int((len(Y)) / 2)): s2 += Y[2 \* i]

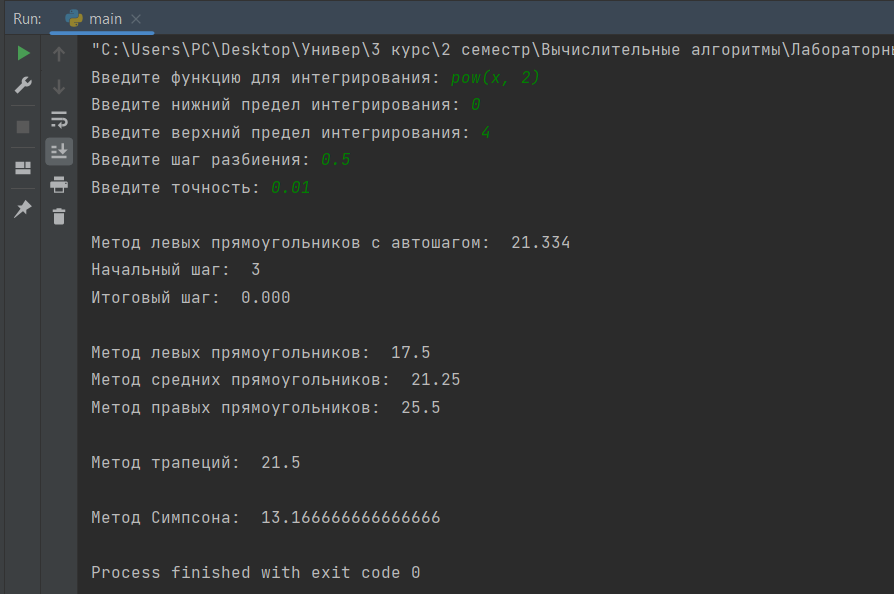
simp = (simp + s1 \* 4 + s2 \* 2) \* step / 3 print("\nМетод Симпсона: ", str(simp))

# Тесты:

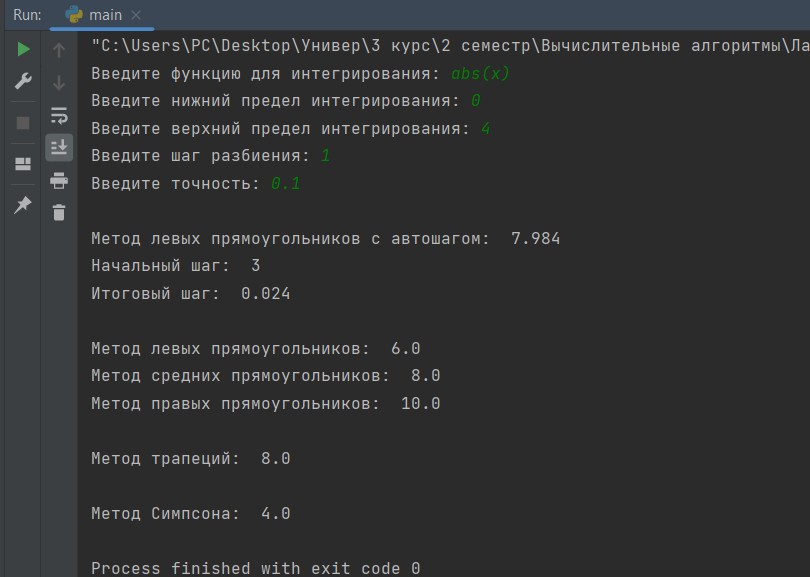
1. Вычисляем интеграл от функции y = x2 от 0 до 4. Шаг для методов с постоянным шагом примем 1, точность для метода с авто шагом 0.1. Результат должен быть приближенно равен 64/3 ≈ 21.33.



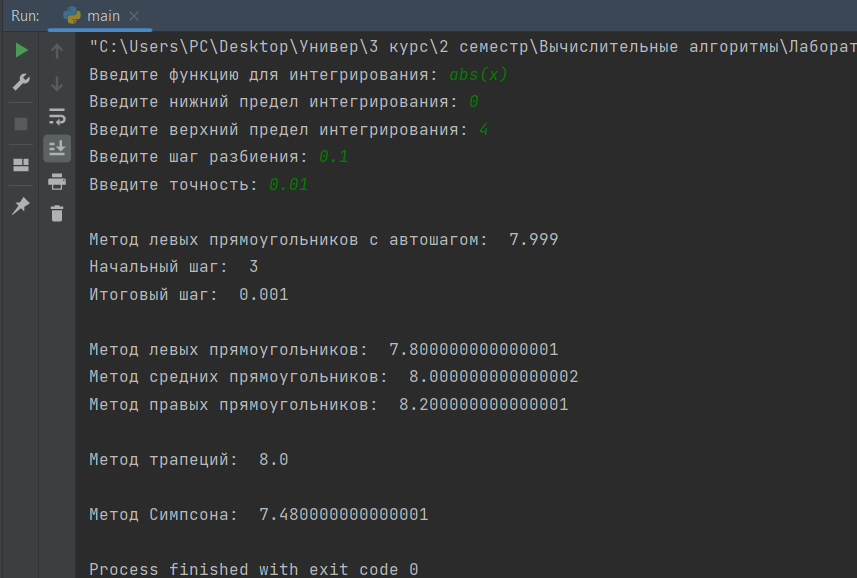
1. Вычисляем интеграл от функции y = x2 от 0 до 4. Шаг для методов с постоянным шагом примем 0.5, точность для метода с авто шагом 0.01. Результат должен быть приближенно равен 64/3 ≈ 21.33.



1. Вычисляем интеграл от функции y = |x| от 0 до 4. Шаг для методов с постоянным шагом примем 1, точность для метода с авто шагом 0.1. Результат должен быть приближенно равен 8.



1. Вычисляем интеграл от функции y = |x| от 0 до 4. Шаг для методов с постоянным шагом примем 0.1, точность для метода с авто шагом 0.01. Результат должен быть приближенно равен 8.



**Вывод:** В ходе проведения вычислительных экспериментов стало ясно, что метод Симпсона дает результаты наиболее далекие от истинного значения, что вероятнее всего было связано с выбором функции для вычисления и довольно большим шагом. Более точными методами оказались метод средних прямоугольников и метод трапеций. Из методов прямоугольников менее точные результаты показал метод левых прямоугольников. При выполнении вычислений с автошагом для метода левых прямоугольников итоговый шаг значительно уменьшается при увеличении требуемой точности, то есть уменьшении значения eps.