|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.04 Программная инженерия**

**Отчет**

|  |
| --- |
| **по лабораторной работе № 5** |

Название:

**Построение и программная реализация алгоритмов численного интегрирования.**

Дисциплина: Вычислительные алгоритмы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ7-46Б |  |  | Нгуен Ань Тхы |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Градов В.М. |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Цель работы**. Получение навыков построения алгоритма вычисления двукратного интеграла с использованием квадратурных формул Гаусса и Симпсона.

**I. Исходные данные.**

Построить алгоритм и программу для вычисления двукратного интеграла при фиксированном значении параметра 

,

где ,

- углы сферических координат.

Применить метод последовательного интегрирования. По одному направлению использовать формулу Гаусса, а по другому - формулу Симпсона.

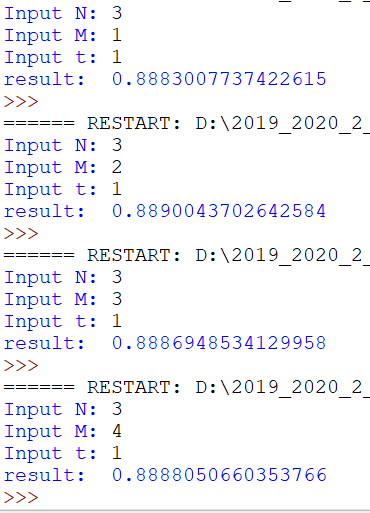
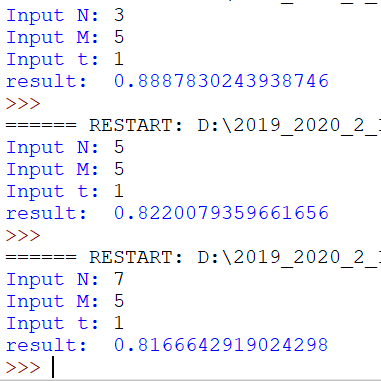
**II. Результат работы программы.**

1. Описать алгоритм вычисления n корней полинома Лежандра n-ой степени  при реализации формулы Гаусса.

2. Исследовать влияние количества выбираемых узлов сетки по каждому направлению на точность расчетов.

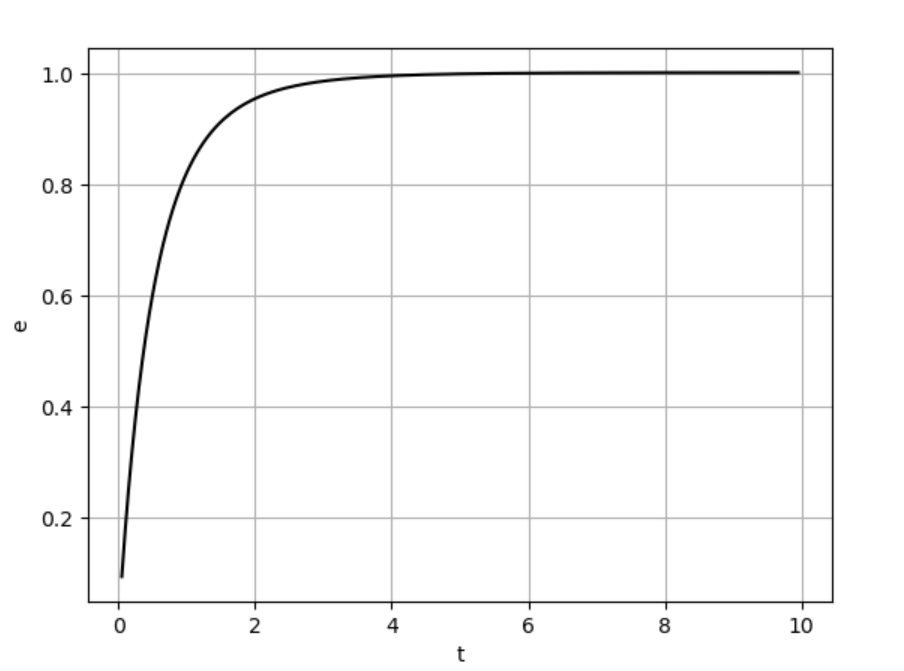
3. Построить график зависимости  в диапазоне изменения =0.05-10. Указать при каком количестве узлов получены результаты.

**III. Пример выполнения прграммы:**

**** ****

При увеличении M результат почти совпадает, что говорит нам о том, что больще влияние оказывает точность внешнего интегрирования.

**график зависимости  в диапазоне изменения =0.05-10:**

****

N = N = 5. Как и было сказано в физическом содержание задачи степень черноты не может быть больше 1.

**IV. Описание алгоритма:**

1. **Квадратная формула Гаусса:**

Пусть интервал вычисляется на стандартном интервале [-1;1]. Квадратная формула:

(5.1)

Запишем полином в виде

k = 0, 1, 2…, 2n – 1 (5.2)

Итак, согласнр (5.2) коэффициенты Ai и узлы ti находятся из системы 2n уравнений

(5.3)

Система (5.30 нелинейная, и ее решение найти довольно трудно. Рассмотрим полиномы Лежандра:

, n = 0, 1, 2,…

Полиномы Лежандра обладают рядом полезных свойств:

1) Pn(1) = 1, Pn(-1) = (-1)n, n = 0, 1, 2…

2)

3) Полином Лежандра Pn(x) имеет n различных и действительных корней, расположенных на интервале [-1, 1] (5.4)

4) Справедливо рекуррентное соотношение

(5.5)

При вычислении интеграла на произвольном интервале [a;b] т.е для применения квадратурной формулы Гаусса необходимо выпольнить преобразование переменной, получим:

= (5.6)

где: , i = 1, 2,…n (5.7)

1. **Другие формулы численного интегрирования:**

F = (5.8)

Построена формула Симпсона:

= (5.13)

Рассмотрим интеграл по прямоугольной области:

, где:

По каждой координате введем сетку узлов. Каждый однократный интеграл вычисляют по квадратным формулам. Конечная формула:

,

где: Ai, Bij – известные постоянные

**V. Ответ на вопросы защита:**

1. В каких ситуациях теоретический порядок квадратурных формул численного интегрирования не достигается.

Если подынтегральная функция не имеет соответствующих произвольных. К примеру, если на отрезке интегрирования не существует 3-я и 4-я производные, то порядок точности формулы Симпсона будет только 2-ой O(h2)

2. Построить формулу Гаусса численного интегрирования при одном узле.

3. Построить формулу Гаусса численного интегрирования при двух узлах.

4. Получить обобщенную кубатурную формулу, аналогичную (6.6) из лекции №6, для вычисления двойного интеграла методом последовательного интегрирования на основе формулы трапеций с **тремя** узлами по каждому направлению.

Где:

**VI. Код программы:**

from numpy.polynomial.legendre import leggauss

import numpy as np

from math import pi, exp, sin, cos

import matplotlib.pyplot as plt

LIMITS = [[0, pi / 2], [0, pi / 2]]

def converts(func2, value):

return lambda y: func2(value, y)

def variable\_conversion(a, b, t):

return (b + a) / 2 + (b - a) \* t / 2

def function(parameter):

return lambda x, y: (4 / pi) \* (1 - exp(-parameter \* 2 \* cos(x) / (1 - (sin(x) \*\* 2) \* (cos(y) \*\* 2)))) \* cos(x) \* sin(x)

def gauss(func, a, b, amounts):

args, coeffs = leggauss(amounts)

res = 0

for i in range(amounts):

res += (b - a) / 2 \* coeffs[i] \* func(variable\_conversion(a, b, args[i]))

return res

def simpson(func, a, b, amounts):

if (amounts < 3 or amounts & 1 == 0):

raise ValueError

h = (b - a) / (amounts - 1)

x = a

res = 0

for i in range((amounts - 1) // 2):

res += func(x) + 4 \* func(x + h) + func(x + 2 \* h)

x += 2 \* h

return res \* (h / 3)

def result(func, n, m, tao):

return simpson(lambda x: gauss(converts(func, x), LIMITS[1][0], LIMITS[1][1], m), LIMITS[0][0], LIMITS[0][1], n)

def main():

N = int(input("Input N: "))

M = int(input("Input M: "))

tao = float(input("Input t: "))

print("result: ", result(function(tao), N, M, tao))

def graph():

plt.grid(True)

plt.xlabel("t")

plt.ylabel("e")

N = 5

M = 5

T = np.arange(0.05, 10, 0.05)

E = [result(function(tao), N, M, tao) for tao in T]

plt.plot(T, E, "k")

plt.show()

main()

graph()