**Лабораторная работа №4**

**Численное интегрирование**

Выполнил студент 2 курса 3 группы ФПМИ

Сараев Владислав Максимович

Минск, 2020

**Теоретические сведения**

Даны функции ,

Необходимо вычислить интеграл используя составную квадратурную формулу трапеций с точность Для оценки погрешности использовать правило Рунге.

Используя систему компьютерной алгебры (в данном случае Wolphram Mathematica) произвести вычисление интеграла (3) и сравнить с ранее полученным.

Для данного вида интегралов построить квадратурную формулу НАСТ с 8 узлами. Используя формулу НАСТ произвести вычисление (3).

Составная квадратурная формула трапеций имеет вид:

Правило Рунге имеет вид: , где для формулы трапеций .

Для вычисление формулы НАСТ , необходимо вычислить .

Для их нахождения необходимо систему .

являются корнями , а .

**Полученное решение**

EXACT = 0.436547 – точное решение интеграла

A = [0.00775384, 0.0761026, 0.231437, 0.331254, 0.251019, 0.0995173, 0.0178684, 0.000920897] – вектор

X = [-0.866595, -0.69679, -0.482143, -0.235601, 0.0275523, 0.291019, 0.538768, 0.757123] – вектор

|  |  |  |  |
| --- | --- | --- | --- |
| Тип квадратурной формулы | Требуемая точность, используемая в правиле Рунге | Достигнутая точность | Количество вычислений подынтегральной функции |
| Составная формула трапеций |  | 0.0001641387122007787 | 12 |
| Составная формула трапеций |  | 2.5035616774138347e-06 | 18 |
| Составная формула трапеций |  | 4.3115348458133695e-08 | 25 |
| НАСТ | - | 1.418600626412747e-07 | 8 |

**Исходный код**

**from** math **import** sin**,** cos**,** e

L **=** **-**1

R **=** 1

EXACT **=** 0.436547

A **=** **[**0.00775384**,** 0.0761026**,** 0.231437**,** 0.331254**,** 0.251019**,** 0.0995173**,** 0.0178684**,** 0.000920897**]**

X **=** **[-**0.866595**,** **-**0.69679**,** **-**0.482143**,** **-**0.235601**,** 0.0275523**,** 0.291019**,** 0.538768**,** 0.757123**]**

**def** f**(**t**):**

**return** sin**(**1 **-** cos**(**e **\*\*** **(**t **\*\*** 5**)))**

**def** ro**(**t**):**

**return** **((**1 **-** t**)** **\*\*** 5**)** **\*** **((**1 **+** t**)** **\*\*** 3**)**

**def** quadrature\_formula**(**n**):**

x **=** L

h **=** **abs(**L **-** R**)** **/** **(**n **-** 1**)**

Q **=** f**(**x**)** **/** 2

**for** k **in** **range(**0**,** n **-** 1**):**

x **+=** h

Q **+=** f**(**x**)** **\*** ro**(**x**)**

x **+=** h

Q **+=** f**(**x**)** **/** 2

Q **\*=** h

**return** Q

faults **=** **[**10 **\*\*** **(-**4**),** 10 **\*\*** **(-**6**),** 10 **\*\*** **(-**8**)]**

**for** fault **in** faults**:**

n **=** 3

answers **=** **[**quadrature\_formula**(**n**)]**

n **\*=** 2

current\_fault **=** 1

**while** current\_fault **>** fault**:**

answers**.**append**(**quadrature\_formula**(**n**))**

current\_fault **=** 1 **/** 3 **\*** **abs(**answers**[len(**answers**)** **-** 1**]** **-** answers**[len(**answers**)** **-** 2**])**

n **\*=** 2

**print(**"answer = " **+** **str(**answers**[len(**answers**)** **-** 1**])** **+** " - " **+** "num = " **+** **str(len(**answers**))** **+** " - " **+** **str(**

**abs(**answers**[len(**answers**)** **-** 1**]** **-** EXACT**)))**

res **=** 0

**for** i **in** **range(**8**):**

res **+=** A**[**i**]** **\*** f**(**X**[**i**])**

**print(**"answer = " **+** **str(**res**)** **+** " - " **+** **str(abs(**res **-** EXACT**)))**