

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики и информатики

**Отчёт по лабораторной работе №4
«Приближенное вычисление интеграла»**

Выполнил:
Гаргома А. О.

Преподаватель:
Горбачёва Ю. Н.

Минск, 2020

Содержание

1	Постановка задачи	1
2	Теория	1
3	Программа	2
4	Результаты работы	4
5	Вывод	5

1 Постановка задачи

Вычислить интеграл $\int_a^b f(x)dx$ с точностью $\varepsilon = 10^{-4}$, используя квадратурные формулы, указанные в варианте задания, и правило Рунге оценки погрешности.

Вычислить интеграл $\int_a^b f(x)dx$ по квадратурной формуле Гаусса с 2, 3 и 4 узлами единичной весовой функции на $[a, b]$.

$$f(x) = \frac{x}{\ln(1+x+x^2)}, \quad a=2, \quad b=3$$

Квадратурные формулы: правых прямоугольников, трапеций, Симпсона.

2 Теория

Формула правых прямоугольников:

$$\int_a^b f(x)dx \approx \sum_{i=1}^n f(x_i)(x_i - x_{i-1})$$

Формула трапеций:

$$\int_a^b f(x)dx \approx \sum_{i=1}^n \frac{f(x_{i-1}) + f(x_i)}{2} (x_i - x_{i-1})$$

Формула Симпсона:

$$\int_a^b f(x)dx \approx \sum_{i=1}^N \frac{f(x_{i-1}) + 4f(\frac{x_{i-1}+x_i}{2}) + f(x_i)}{6} (x_i - x_{i-1})$$

Погрешность вычисления значения интеграла при числе шагов, равном $2n$ определяется по формуле Рунге:

$$\Delta_{2n} \approx \Theta |I_{2n} - I_n|$$

Для формулы правых прямоугольников $\Theta = 1$, для формулы трапеций $\Theta = \frac{1}{3}$, для формулы Симпсона $\Theta = \frac{1}{15}$.

Укажем квадратурные формулы Гаусса для весовой функции $p \equiv 1$ и отрезка интегрирования $[-1, 1]$.

Формула Гаусса для двух узлов:

$$I \approx f(-0.5773502692) + f(0.5773502692)$$

Формула Гаусса для трёх узлов:

$$I \approx \frac{5}{9}f(-0.7745966692) + \frac{8}{9}f(0) + \frac{5}{9}f(0.7745966692)$$

Формула Гаусса для четырёх узлов:

$$I \approx 0.3478548451f(-0.8611363116) + 0.6521451549f(-0.3399810436) + \\ 0.6521451549f(0.3399810436) + 0.3478548451f(0.8611363116)$$

Для применения формул Гаусса на отрезке $[a, b]$, можно воспользоваться линейной заменой

$$x = \frac{a+b}{2} + \frac{b-a}{2}t$$

и домножить результат на

$$\frac{b-a}{2}$$

Значения узлов x_i метода Гаусса по n точкам являются корнями полинома Лежандра степени n . Значения весов вычисляются по формуле

$$a_i = \frac{2}{(1-x_i^2)[P'_n(x_i)]^2}$$

где P'_n – первая производная полинома Лежандра.

3 Программа

```
import numpy as np
import math
```

```
def f(x):
    return 1/(np.log(1+x+x**2))
```

```
a, b = 2, 3
```

```
eps = 10**(-4)
```

```
runge_coeffs = {'right_rectangles_method': 1, 'trapezium_method':
    1/3,
    'simpsons_method': 1/15}
```

```
nodes = {2: [-0.55773502692, 0.55773502692],
    3: [-0.7745966692, 0, 0.7745966692],
    4: [-0.8611363116, -0.3399810436, 0.3399810436,
    0.8611363116]}
```

```

gauss_coeffs = {2: [1, 1],
                 3: [5/9, 8/9, 5/9],
                 4: [0.3478548451, 0.6521451549, 0.6521451549,
                    0.3478548451]}

# преобразование  $[-1, 1] \rightarrow [a, b]$ 
def transform(x):
    return (a + b) / 2 + (b - a) * x / 2

# погрешность по правилу Рунге
def runge(prev, curr, theta):
    return np.abs(curr - prev) * theta

def right_rectangles_method(n):
    xs = np.linspace(a, b, n+1)
    h = xs[1:] - xs[:-1] #  $h_i = x_i - x_{i-1}$ ,  $i = 1, \dots$ 
    return np.sum(f(xs[1:]) * h) # сумма  $f(x_i) * h_i$ 

def trapezium_method(n):
    xs = np.linspace(a, b, n+1)
    h = xs[1:] - xs[:-1]
    return np.sum((f(xs[:-1]) + f(xs[1:]))) / 2 * h

def simpsons_method(n):
    xs = np.linspace(a, b, n+1)
    h = xs[1:] - xs[:-1]
    return np.sum((f(xs[:-1]) + 4 * f((xs[:-1] + xs[1:])/2) +
                  + f(xs[1:]))) * h / 6

def gauss_formula(n):
    result = 0
    for (xi, ai) in zip(nodes[n], gauss_coeffs[n]):
        result += ai * f(transform(xi))
    return result * (b-a)/2

def calculate(method, method_name):
    theta = runge_coeffs[method_name]
    n = 1
    h = b - a
    curr = method(n)
    error = math.inf

```

```

    print(f'& $h = {h:.6f}$ & $I_h={curr:.6f}$ & \\\cline
{{2-4}} ')
    while error > eps:
        n *= 2
        h /= 2
        prev, curr = curr, method(n)
        error = runge(prev, curr, theta)
        print(f'& $h/{n} = {h:.6f}$ & $I_{{h/{n}}}={curr:.6f}$ & $R_{{h/{n}}}=\\
{error:.6f}$ \\\cline{{2-4}} ')

calculate(right_rectangles_method, 'right_rectangles_method')
calculate(trapezium_method, 'trapezium_method')
calculate(simpsons_method, 'simpsons_method')

for i in range(2, 5):
    print(f'{{i}}:\t{gauss_formula(i):.10f}')

```

4 Результаты работы

Формула	Шаг	Значение	Погрешность
Правых прямоугольн.	$h = 1.000000$	$I_h = 0.389871$	
	$h/2 = 0.500000$	$I_{h/2} = 0.414497$	$R_{h/2} = 0.024626$
	$h/4 = 0.250000$	$I_{h/4} = 0.428352$	$R_{h/4} = 0.013855$
	$h/8 = 0.125000$	$I_{h/8} = 0.435688$	$R_{h/8} = 0.007336$
	$h/16 = 0.062500$	$I_{h/16} = 0.439459$	$R_{h/16} = 0.003772$
	$h/32 = 0.031250$	$I_{h/32} = 0.441371$	$R_{h/32} = 0.001912$
	$h/64 = 0.015625$	$I_{h/64} = 0.442334$	$R_{h/64} = 0.000962$
	$h/128 = 0.007812$	$I_{h/128} = 0.442817$	$R_{h/128} = 0.000483$
	$h/256 = 0.003906$	$I_{h/256} = 0.443058$	$R_{h/256} = 0.000242$
	$h/512 = 0.001953$	$I_{h/512} = 0.443179$	$R_{h/512} = 0.000121$
Трапеций	$h = 1.000000$	$I_h = 0.451885$	
	$h/2 = 0.500000$	$I_{h/2} = 0.445504$	$R_{h/2} = 0.002127$
	$h/4 = 0.250000$	$I_{h/4} = 0.443855$	$R_{h/4} = 0.000549$
	$h/8 = 0.125000$	$I_{h/8} = 0.443439$	$R_{h/8} = 0.000139$
	$h/16 = 0.062500$	$I_{h/16} = 0.443335$	$R_{h/16} = 0.000035$
Симпсона	$h = 1.000000$	$I_h = 0.443377$	
	$h/2 = 0.500000$	$I_{h/2} = 0.443306$	$R_{h/2} = 0.000005$

Количество узлов	Приближенное значение интеграла
2	0.4429706518
3	0.4432999292
4	0.4433004909

5 Вывод

Из количества итераций методов в таблице выше можно сделать вывод, что сходимости этих методов можно расположить по возрастанию: метод правых прямоугольников $<$ метод трапеций $<$ метод Симпсона. Квадратуры Гаусса также позволяют быстро получить ответ с высокой точностью, но для оценки погрешности требуется много вычислений.