Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра програмного забезпечення

ЗВІТ

Про виконання лабораторної роботи № 10

«Чисельні методи інтегрування»
з дисципліни «Чисельні методи»

Лектор:

Мельник Н.Б.

Виконав:

студент групи ПЗ-11

Ясногородський Н.В.

Прийняла:

Мельник Н.Б.

«___» ______ 2022 р.

$\sum$ = ____ ___________

Львів – 2022

Тема роботи: Чисельні методи інтегрування.
Мета роботи: Ознайомлення на практиці з методами чисельного інтегрування.

Теоретичні відомості

Метод прямокутників

Найпростішим методом наближеного обчислення інтеграла є метод прямокутників, суть якого зводиться до знаходження означеного інтеграла як суми площ n прямокутників висотою $f(x_i)$ та основою $h = x_i$, отриманих шляхом розбиття відрізка інтегрування [a,b] на n рівних частин.

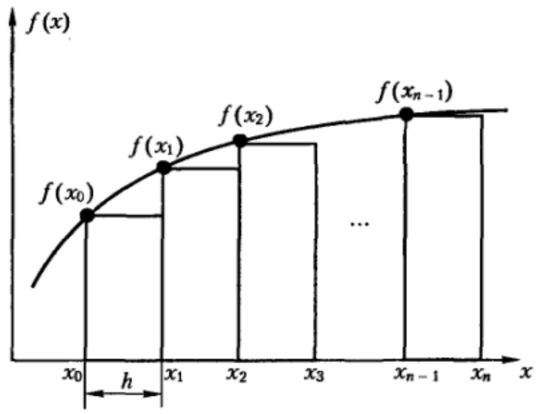


Рис.1 Метод лівих прямокутників

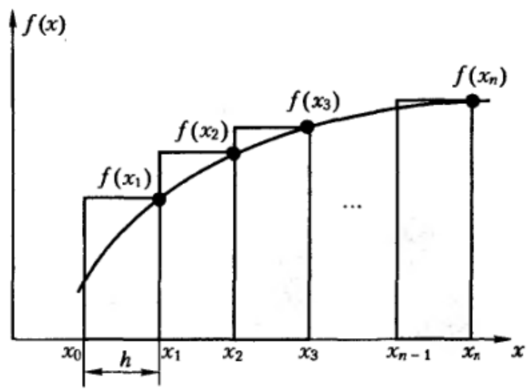


Рис.2 Метод правих прямокутників

Формула лівих прямокутників:

$$I_л = \int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} f(x_i)$$

Формула правих прямокутників:

$$I_{np} = \int_a^b f(x)dx \approx h\sum_{i=1}^{n} f(x_i)$$

Де $h = \dfrac{b - a}{n}$

Однак ці методи не є дуже точними, оскільки в першому є нестача, а в другому надлишок інтегралу. Тому існує метод середніх прямокутників.

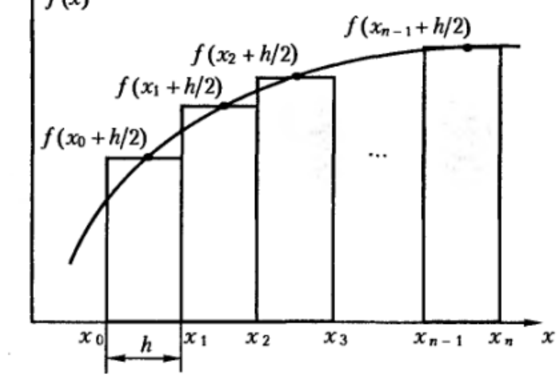


Рис.3 Метод середніх прямокутників.

Формула середніх прямокутників:

$$I_{сер} = \int_a^b f(x)dx \approx h\sum_{i=0}^{n-1} f(x_i + \frac{h}{2})$$

Метод трапецій

Метод трапецій полягає в тому, що відрізок інтегрування [a,b] розбивають на n рівних відрізків, а криву, описану підінтегральну функцією f (x), замінюють на кожному із цих відрізків кусково-лінійною функцією φ(x), отриманою стягуванням хорд, які проходять через точки $(x_{i-1}, f(x_{i-1}))$ $та$ $(x_i, f(x_i))$ $(i = 1, n)$.

Значення інтеграла знаходять як суму площ S_i $(i = 0, n)$ прямокутних трапецій з висотою $h = \frac{b-a}{n}$.

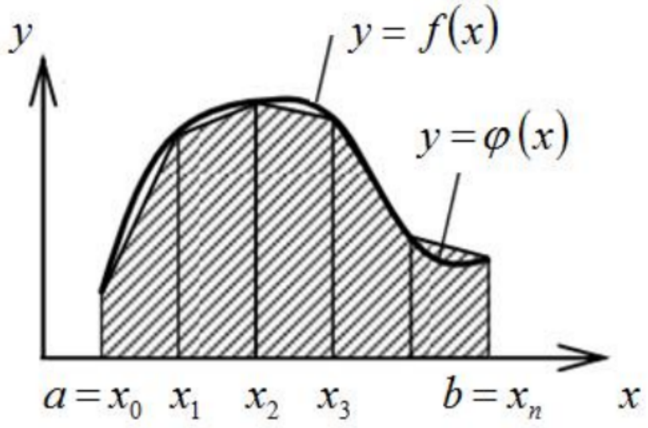


Рис.4 метод трапецій

Площу кожної i-ої елементарної трапеції визначають за формулою

$$S_i = h\frac{f(x_i) + f(x_{i+1})}{2}$$

Відповідно отримуємо таку формулу трапецій для обчислення інтегралу:

$$I_{тр} = \int_a^b f(x)dx \approx h(\frac{f(x_0) + f(x_n)}{2} + \sum_{i=1}^{n-1} f(x_i))$$

Метод Сімпсона

Даний метод полягає в тому, що криву, описану підінтегральною функцією f (x) , на елементарних відрізках заміняють параболою.

Поділимо відрізок інтегрування [a,b] на парну кількість n рівних частин з кроком

$h = \frac{b - a}{n}$. На кожному елементарному відрізку
$[x_0, x_2], [x_2, x_4], \ldots, [x_{i-1}, x_{i+1}], \ldots, [x_{n-2}, x_n]$ підінтегральну функцію f (x) замінимо інтерполяційним поліномом другого степеня (квадратичною параболою). Тоді обчислення означеного інтеграла зводиться до обчислення суми площ S_i, (i =1,n) криволінійних трапецій.

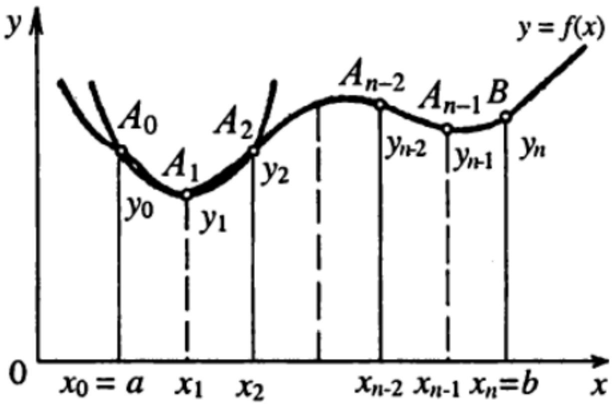


Рис.5 Метод Сімпсона

Площу S_i кожної елементарної криволінійної трапеції визначають за формулою Сімпсона

$$S_i = \frac{h}{3}(f(x_i) + 4f(x_{i+1}) + f(x_{i+2}))$$

Формула Сімпсона:

$$\int_a^b f(x)dx \approx \frac{h}{3}(f(x_0) + 4\sum_{i=1}^{n} f(x_{2i-1}) + 2\sum_{i=1}^{n-1} f(x_{2i}) + f(x_{2n}))$$

Індивідуальне завдання

Складти програму чисельного інтегрування у відповідності до варіанту:

1) Методом лівих, правих та центральних прямокутників;
2) Методом трапецій;
3) Методом Сімпсона

Варіант 10.

$$10. \int_0^3 \sqrt{x} \ln(1 + \sqrt[3]{x^2})\, dx;$$

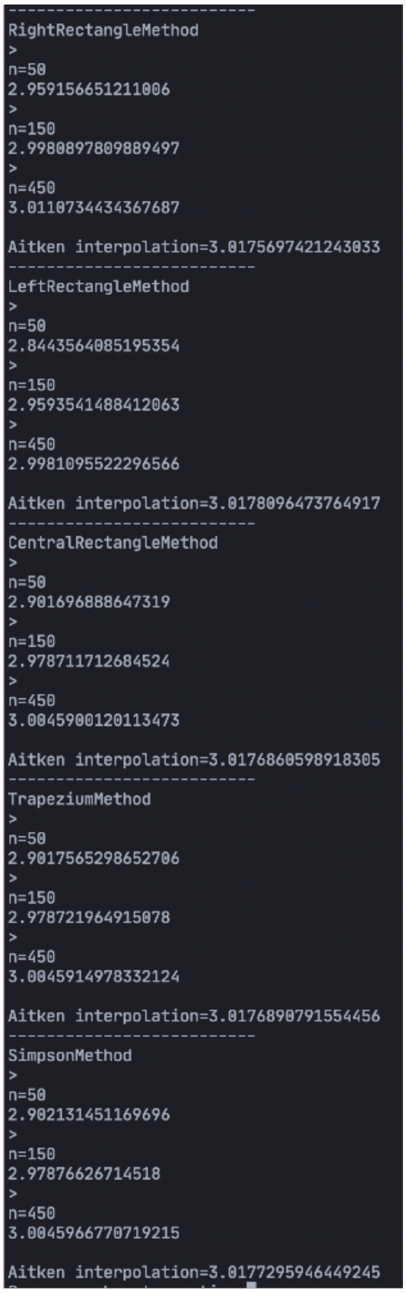
Результат виконання роботи

```
import json
import string

import numpy as np


def f(x):
    return np.sqrt(x) * np.log(1 + np.cbrt(x**2))


a = 0
b = 3
N = 50


def left_rectangle_method(f, a, b, n=N):
    h = (b - a) / n

    def f_xi(i):
        return f(a + i * h)

    return h * np.sum([f_xi(i) for i in range(n - 1)])


def right_rectangle_method(f, a, b, n=N):
    h = (b - a) / n

    def f_xi(i):
        return f(a + i * h)

    return h * np.sum([f_xi(i) for i in range(1, n)])


def central_rectangle_method(f, a, b, n=N):
    h = (b - a) / n

    def f_xi(i):
        return f(a + i * h)

    return h * np.sum([f_xi(i + 0.5) for i in range(n - 1)])


def trapezium_method(f, a, b, n=N):
    h = (b - a) / n

    def f_xi(i):
```

```
        return f(a + i * h)

    trapezium = h * np.sum([(f_xi(i) + f_xi(i + 1)) / 2 for i in range(n - 1)])

    return trapezium


def simpson_method(f, a, b, n=N):
    h = (b - a) / n

    def f_xi(i):
        return f(a + i * h)

    simpson = (h / 3) * (
        f_xi(0)
        + f_xi(n)
        + 4 * np.sum([f_xi(i - 1) for i in range(1, n) if i % 2])
        + 2 * np.sum([f_xi(i) for i in range(1, n - 1) if i % 2])
    )

    return simpson


def snake_case_to_pascal_case(str):
    return string.capwords(str.replace("_", " ")).replace(" ", "")


if __name__ == "__main__":
    for method in [
        right_rectangle_method,
        left_rectangle_method,
        central_rectangle_method,
        trapezium_method,
        simpson_method,
    ]:
        print("--------------------------")
        print(snake_case_to_pascal_case(method.__name__))
        results = []
        for i in range(3):
            n = N * (3**i)
            print(">")
            print(f"n={n}")
            res = method(f, a, b, n)
            results.append(res)
            print(json.dumps(res, indent=4))

        better_res = results[0] - ((results[0] - results[1]) ** 2) / (
            results[0] - 2 * results[1] + results[2]
        )
```

```
print(f"\nAitken interpolation={better_res}")
```

Результат виконання програми


```
-------------------------
RightRectangleMethod
>
n=50
2.959156651211006
>
n=150
2.9980897809889497
>
n=450
3.0110734434367687

Aitken interpolation=3.0175697421243033
-------------------------
LeftRectangleMethod
>
n=50
2.8443564085195354
>
n=150
2.9593541488412063
>
n=450
2.9981095522296566

Aitken interpolation=3.0178096473764917
-------------------------
CentralRectangleMethod
>
n=50
2.901696888647319
>
n=150
2.978711712684524
>
n=450
3.0045900120113473

Aitken interpolation=3.0176860598918305
-------------------------
TrapeziumMethod
>
n=50
2.9017565298652706
>
n=150
2.978721964915078
>
n=450
3.0045914978332124

Aitken interpolation=3.0176890791554456
-------------------------
SimpsonMethod
>
n=50
2.902131451169696
>
n=150
2.97876626714518
>
n=450
3.0045966770719215

Aitken interpolation=3.0177295946449245
```

Висновки

Виконуючи лабораторну роботу №10, я ознайомився на практиці з методами чисельного інтегрування.