МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное образовательное учреждение высшего образования **«Национальный исследовательский университет ИТМО»**

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №3

‘ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА’

Вариант №25

Студент:
Хоанг Ван Куан
Группа P3266

Преподаватель:
Машина Екатерина Александровна

Санкт-Петербург, 2024

1. Цель работы

Найти приближенное значение определенного интеграла с требуемой точностью различными численными методами

2. Порядок выполнения работы

<u>Обязательное задание</u>

- 1 часть: Вычислительная реализация задачи
  Задание:
  1. Вычислить интеграл

$$\int_0^2 (2x^3 - 4x^2 + 6x - 25)dx$$

  2. Вычислить интеграл по формуле Ньютона – Котеса при n = 8
  3. Вычислить интеграл по формулам средних прямоугольников, трапеций и Симпсона при n = 10
  4. Сравнить результаты с точным значением интеграла
  5. Определить относительную погрешность вычислений для каждого метода

- 2 часть: Программная реализация

<u>Необязательное задание</u>

1. Установить сходимость рассматриваемых несобственных интегралов 2 рода (2-3 функции). Если интеграл - расходящийся, выводить вообщение: «Интеграл не существует»;
2. Если интеграл сходящийся, реализовать в программе вычисление несобственных интегралов 2 рода (заданными численными методами)
3. Рассмотреть случаи, когда подынтегральная функция терпит бесконечный разрыв: 1) в точке a, 2) в точке b, 3) на отрезке интегрирования.

3. Рабочие формулы

1 – Формула Ньютона – Котеса

$$\int_a^b f(x) \approx \int_a^b L_n(x)dx = \sum_{i=0}^{n} f(x_i)c_n^i$$

2 – Формула метода пряугольников

$$\int_a^b f(x) \approx S_n = \sum_{i=0}^{n} f(\xi_i)\Delta x_i$$

3 – Формула метода трапеций

$$\int_a^b f(x) \approx \frac{h}{2}\left(y_0 + y_n + 2\sum_{i=1}^{n-1} y_i\right)$$

4 – Формула метода Симпсона

$$\int_a^b f(x) \approx \frac{h}{3}(y_0 + 4(y_1 + y_3 + \cdots + y_{n-1}) + 2(y_2 + y_4 + \cdots + y_{n-2}) + y_n)$$

4. Вычислительная часть

1 – Вычислить интеграл

$$\int_0^2 (2x^3 - 4x^2 + 6x - 25)dx = \frac{1}{2}x^4 - \frac{4}{3}x^3 + 3x^2 - 25x\Big|_0^2 = \frac{-122}{3} = -40.(6)$$

2 – Вычислить интеграл по формуле Ньютона – Котеса при n = 8

Вычиляем коэффициенты Ньютона – Котеса

$$c_8^0 = c_8^8 = \frac{989\,.(2-0)}{28350} = \frac{989}{14175}$$

$$c_8^1 = c_8^7 = \frac{5888\,.(2-0)}{28350} = \frac{5888}{14175}$$

$$c_8^2 = c_8^6 = -\frac{928\,.(2-0)}{28350} = -\frac{928}{14175}$$

$$c_8^3 = c_8^5 = \frac{10496\,.(2-0)}{28350} = \frac{10496}{14175}$$

$$c_8^0 = -\frac{4540\,.(2-0)}{28350} = -\frac{908}{2835}$$

$$\int_0^2 (2x^3 - 4x^2 + 6x - 25)dx \approx \sum_{i=0}^{8} f(x_i)c_n^i$$

$$\approx c_8^0 f(0) + c_8^1 f\left(\frac{2}{8}\right) + c_8^2 f\left(\frac{4}{8}\right) + c_8^3 f\left(\frac{6}{8}\right) + c_8^4 f\left(\frac{8}{8}\right) + c_8^5 f\left(\frac{10}{8}\right) + c_8^6 f\left(\frac{12}{8}\right) + c_8^7 f\left(\frac{14}{8}\right) + c_8^8 f(2)$$

$$\approx -40.667$$

3 – Вычислить интеграл по формулам средних прямоугольников, трапеций и Симпсона при n = 10

- Метод средних прямоугольников

Разобьем отрезок интегрирования на 10 равных частей $n = 10, h = \dfrac{b - a}{n} = 0.2$

i	0	1	2	3	4	5	6	7	8	9	10

x_i	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
y_i	-25	-23.944	-23.112	-22.408	-21.736	-21	-20.104	-18.952	-17.448	-15.496	-13
$x_{i-\frac{1}{2}}$		0.1	0.3	0.5	0.7	0.9	1.1	1.3	1.5	1.7	1.9
$y_{i-\frac{1}{2}}$		-24.438	-23.506	-22.75	-22.074	-21.382	-20.578	-19.566	-18.25	-16.534	-14.322

По формулам левых, правых и средних прямоугольников получим:

$$I_{\text{прав}} = h\sum_{i=1}^{n} y_i = 0.2 * -197.2 = -39.44$$

$$I_{\text{лев}} = h\sum_{i=1}^{n} y_{i-1} = 0.2 * -209.2 = -41.84$$

$$I_{\text{сред}} = h\sum_{i=1}^{n} y_{i-\frac{1}{2}} = 0.2 * -203.4 = -40.68$$

Погрешность в вычислении интеграла составляет

$$\Delta I_{\text{прав}} = I - I_{\text{прав}} = -40.(6) - (-39.44) = -1.227\ (\approx 3.016\%)$$

$$\Delta I_{\text{лев}} = I - I_{\text{лев}} = -40.(6) - (-41.84) = 1.173\ (\approx 2.885\%)$$

$$\Delta I_{\text{сред}} = I - I_{\text{сред}} = -40.(6) - (-40.68) = 0.01(3)\ (\approx 0.033\%)$$

- Метод трапеций

Разобьем отрезок интегрирования на 10 равных частей $n = 10, h = \dfrac{b - a}{n} = 0.2$

$$I_{\text{трап}} = h.\left(\frac{y_0 + y_n}{2} + \sum_{i=1}^{n-1} y_i\right) = 0,2.\left(\frac{-25 + 15.496}{2} - 184.2\right) = -40.64$$

Погрешность в вычислении интеграла составляет

$$\Delta I_{\text{трап}} = I - I_{\text{трап}} = -40.(6) - (-40.64) = 0.02(6)\ (\approx 0.066\%)$$

i	0	1	2	3	4	5	6	7	8	9	10
x_i	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2

y_i	-25	-23.944	-23.112	-22.408	-21.736	-21	-20.104	-18.952	-17.448	-15.496	-13

- Метод Симпсона

Разобьем отрезок интегрирования на 10 равных частей $n = 10, h = \frac{b-a}{n} = 0.2$

$$I = \frac{h}{3}[y_0 + 4(y_1 + y_3 + \cdots + y_9) + 2(y_2 + y_4 + \cdots + y_8) + y_{10}]$$

$$= \frac{0.2}{3}(-25 - 407.2 - 164.8 - 13) = -40.(6)$$

i	0	1	2	3	4	5	6	7	8	9	10
x_i	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
y_i	-25	-23.944	-23.112	-22.408	-21.736	-21	-20.104	-18.952	-17.448	-15.496	-13

5. Листинг программы

Программы для методов

```
private double LeftRectangleMethod(double a, double b, double h){
    double res = 0;
    for(double i = a; i < b - h/2; i += h) res += h * apply(i);
    return res;
}
private double RightRectangleMethod(double a, double b, double h){
    double res = 0;
    for(double i = a + h; i < b + h/2; i += h) res += h * apply(i);
    return res;
}
private double MiddleRectangleMethod(double a, double b, double h){
    double res = 0;
    for(double i = a + h/2; i < b; i += h) res += h * apply(i);
    return res;
}
private double TrapezoidMethod(double a, double b, double h){
    double sum = 0;
    for(double i = a + h; i < b - h/2; i += h)
        sum += apply(i);
    return h*((apply(a) + apply(b))/2 + sum);
}
private double SimpsonMethod(double a, double b, double h){
    double sum1 = 0;
    double sum2 = 0;
    for(double i = a + h; i < b - h/2; i += 2*h) sum1 += apply(i);
    for(double i = a + 2*h; i < b - h/2; i += 2*h) sum2 += apply(i);
    return h/3*(apply(a) + 4*sum1 + 2*sum2 + apply(b));
}
```

Программы для проверки погрешности

```java
private double RungeRule(double res, double res2){
    return Math.abs(res2 - res)/(Math.pow(2,k) - 1);
}
private double supMethod(double a, double b, double h){
    switch (MethodNum){
        case 1:
            k = 2;
            return LeftRectangleMethod(a, b, h);
        case 2:
            k = 2;
            return RightRectangleMethod(a, b, h);
        case 3:
            k = 2;
            return MiddleRectangleMethod(a, b, h);
        case 4:
            k = 2;
            return TrapezoidMethod(a, b, h);
        default:
            k = 4;
            return SimpsonMethod(a, b, h);
    }
}
private Result subResul(double a, double b){
    Result result = new Result();
    result.n = nmin;
    do{
        result.res = supMethod(a,b,Math.abs(b-a)/result.n);
        result.n *= 2;
        result.res2 = supMethod(a,b,Math.abs(b-a)/ result.n);
    } while(Math.abs(result.res2 - result.res) > epsilon);
    return result;
}
```

Дополнительное задание

```java
private double c; // Особая точка подынтегральной функции
public void resultImproperIntegral(){
    Timer timer = new Timer();
    timer.schedule(new TimerTask() {
        @Override
        public void run() {
            System.out.println("Интеграл не существует");
            System.exit(0);
        }
    }, 3000);
    if(a <= c && c <= b){
        Result res = new Result();
        try{
            double e = 0.0001;
            if(a == c) res = subResul(a + e,b);
            else if(b == c) res = subResul(a, b - e);
            else {
                Result res1 = subResul(a,c - e);
                Result res2 = subResul(c + e,b);
                res.res = res1.res + res2.res;
                res.res2 = res1.res2 + res2.res;
                res.n = Math.max(res1.n, res2.n);
            }
        } catch (Exception e){
            System.out.println("Интеграл не существует");
            return;
        }
        res.print();
        System.exit(0);
```

```
    }
    else{
        resultIntegral();
        System.exit(0);
    }

}
```

6. Результаты выполнения программы
Обязательное здание

```
----------------------------------------------------------
ЧАСТЬ 1: ОБЯЗАТЕЛЬНОЕ ЗАДАНИЕ
Взять исходные данные из файла (+) или ввести с клавиатуры (-)?
Режим ввода:
+
Значение интеграла: 2.314609374999994
Число разбиения: 80
Погрешность по правилу Рунге: 0.003118489583433757
```

```
Попробуйте с другими методами (+/-) ?
Режим ввода:
+
Выберите функцию
 1) Метод левых прямоугольников
 2) Метод средних прямоугольников
 3) Метод правых прямоугольников
 4) Метод трапеций
 5) Метод Сипсона
Выберите метод из списка
4
Значение интеграла: 2.34
Число разбиения: 5
Погрешность по правилу Рунге: 0.0016666666666661871
```

```
Режим ввода:
-
Выберите функцию
 1) 2x³ - 4x² + 6x -25
 2) 1/10x⁴ + 1/5x² - 7
 3) x³ - 2x² - 5x + 7
 4) x²
Выберите функцию из списка
4
Выберите функцию
 1) Метод левых прямоугольников
 2) Метод средних прямоугольников
 3) Метод правых прямоугольников
 4) Метод трапеций
 5) Метод Сипсона
Выберите метод из списка
1
Введите пределы интегрирования:
1 2
Введите точность вычисления:
0.01
Введите начальное значение числа разбиения:
5
Значение интеграла: 2.314609374999994
Число разбиения: 80
Погрешность по правилу Рунге: 0.003118489583343757
```

Дополнительное задание

```
------------------------------------------------------------
ЧАСТЬ 2: НЕОБЯЗАТЕЛЬНОЕ ЗАДАНИЕ
Взять исходные данные из файла (+) или ввести с клавиатуры (-)?
Режим ввода:
+
Интеграл не существует
```

```
Выберите функцию
 5) 1/x²
 6) 1/√x
 7) 1/(1-x)
Выберите функцию из списка
6
Выберите функцию
 1) Метод левых прямоугольников
 2) Метод средних прямоугольников
 3) Метод правых прямоугольников
 4) Метод трапеций
 5) Метод Сипсона
Выберите метод из списка
1
Введите пределы интегрирования:
0 1
Введите точность вычисления:
0.01
Введите начальное значение числа разбиения:
5
Значение интеграла: 1.991051512821941
Число разбиения: 5120
Погрешность по правилу Рунге: 0.0019464219747543272
```

7. Выводы

В результате выполнения данной лабораторной работой я познакомился с численными методами интегрирования и реализовал метод прямоугольников, метод трапеций и метод Симпсона на языке программирования Java, закрепив знания.