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

Факультет программной инженерии и компьютерной техники

Лабораторная работа №3

Вариант 7

Студент: Кулагин Вячеслав Дмитриевич
Преподаватель: Наумова Надежда Александровна

Группа: P3209

Санкт-Петербург
2025

Содержание

1 Цели работы

Найти приближенное значение определенного интеграла с требуемой точностью различными численными методами.

2 Вычислительная реализация задачи

2.1 Решение различными методами

1. Точное значение
$$\int_0^2 (4x^3 - 5x^2 + 6x - 7)dx = \left(x^4 - \frac{5x^3}{3} + 3x^2 - 7x\right)|_0^2 = \frac{2}{3} \approx 0.667$$

2. По формуле Ньютона–Котеса ($n = 6$)
$$\int_0^2 (4x^3 - 5x^2 + 6x - 7)dx = \frac{41 \cdot 2}{840} f(0) + \frac{41 \cdot 2}{840} f(2) + \frac{216 \cdot 2}{840} f(0.333) + \frac{216 \cdot 2}{840} f(1.667) +$$
$$+ \frac{27 \cdot 2}{840} f(0.667) + \frac{27 \cdot 2}{840} f(1.333) + \frac{272 \cdot 2}{840} f(1) =$$
$$= -0.683 + 1.659 - 2.782 + 3.928 - 0.259 + 0.102 - 1.295 = 0.669$$

3. По формуле средних прямоугольников
Разбиение: $h = \frac{b-a}{n} = \frac{2}{10} = 0.2$

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	-7	-5.968	-5.144	-4.336	-3.352	-2	-0.088	2.576	6.184	10.928	17
$x_{i-0.5}$		0.1	0.3	0.5	0.7	0.9	1.1	1.3	1.5	1.7	1.9
$y_{i-0.5}$		-6.446	-5.542	-4.75	-3.878	-2.734	-1.126	1.138	4.25	8.402	13.786

Получаем: $I = h \sum_{i=0}^{n} f(y_{i-0.5}) = 0.62$

4. По формуле трапеций:
Используем ту же таблицу, считаем по трапециям:
$$I = \frac{h}{2}\left(y_0 + y_n + 2\sum_{i=1}^{n-1} y_i\right) = 0.76$$

5. По формуле Симпсона: Используем ту же таблицу, считаем по Симпсону:
$$I = \frac{h}{3}(y_0 + 4(y_1 + y_3 + ... + y_{n-1}) + 2(y_2 + y_4 + ...y_{n-2}) + y_n) = 0.667$$

2.2 Сравнение результатов

Почти все полученные результаты похожи на точное значение интеграла. Относительная погрешность:

- Формула Ньютона-Котеса: 0.002
- Формула средних прямоугольников: 0.047
- Формула трапеций: 0.093
- Формула Симпсона: 0.000 (самый точный результат)

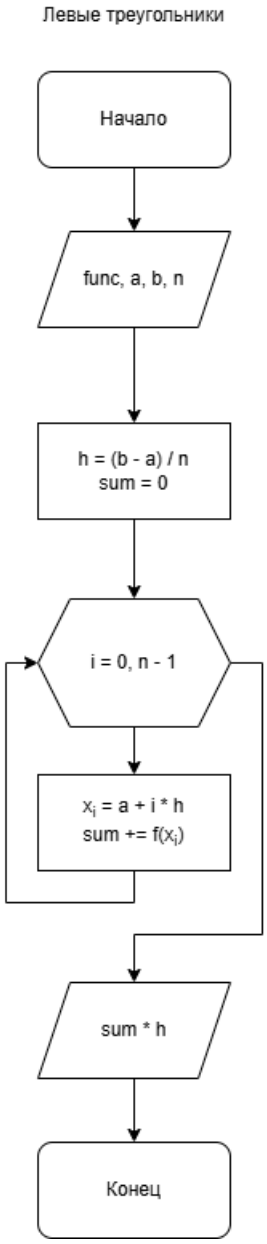
3 Программная реализация

3.1 Метод левых прямоугольников

```
private static double rectangleLeft(FunctionEvaluator func,
                                    double a, double b, int n) {
    double h = (b - a) / n;
    double sum = 0.0;

    for (int i = 0; i < n; i++) {
        double x = a + i * h;
        sum += func.evaluate(x);
    }

    return sum * h;
}
```



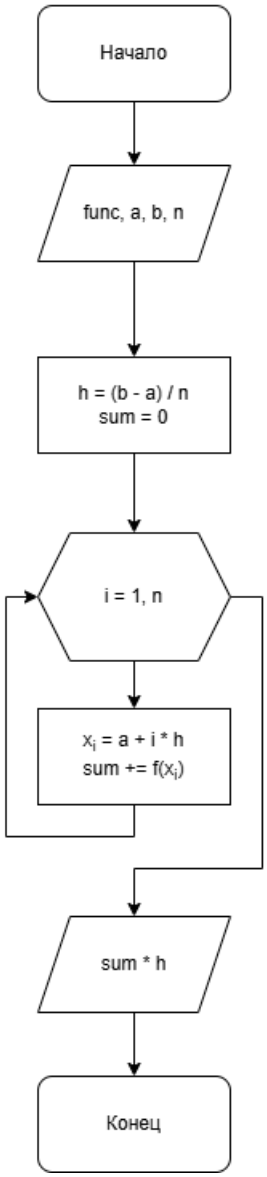
3.2 Метод левых прямоугольников

```
private static double rectangleRight(FunctionEvaluator func,
                                     double a, double b, int n) {
    double h = (b - a) / n;
    double sum = 0.0;

    for (int i = 1; i <= n; i++) {
        double x = a + i * h;
        sum += func.evaluate(x);
    }

    return sum * h;
}
```

Правые треугольники



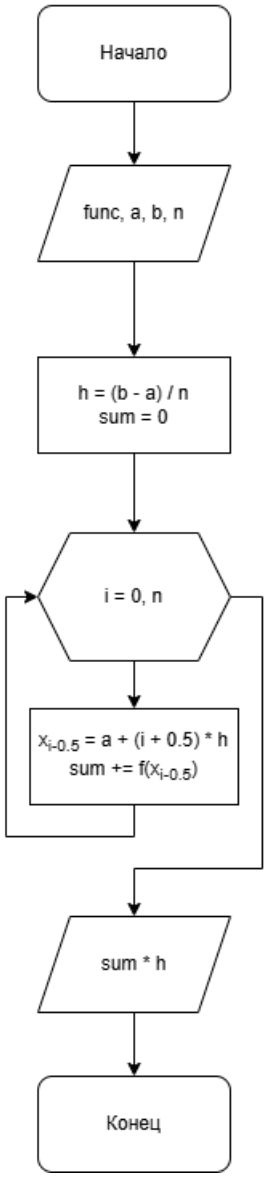
3.3 Метод средних прямоугольников

```
private static double rectangleMid(FunctionEvaluator func,
                                   double a, double b, int n) {
    double h = (b - a) / n;
    double sum = 0.0;

    for (int i = 0; i < n; i++) {
        double x = a + (i + 0.5) * h;
        sum += func.evaluate(x);
    }

    return sum * h;
}
```

Средние треугольники

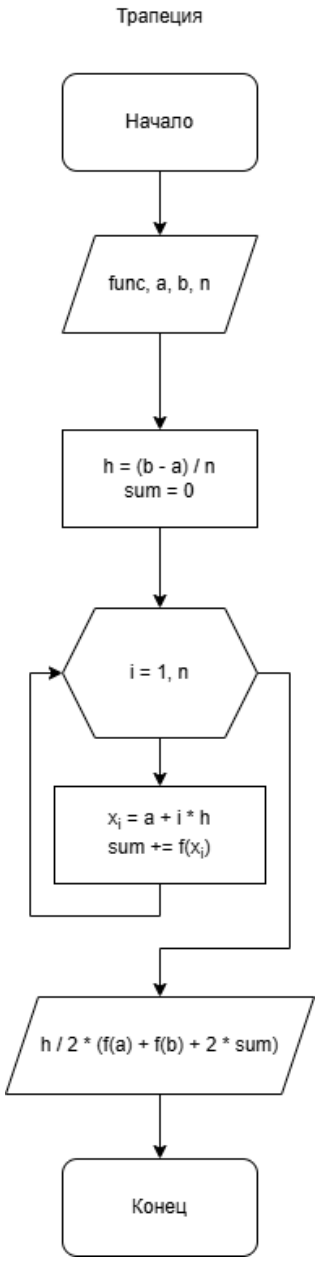


3.4 Метод трапеций

```
private static double trapezoid(FunctionEvaluator func,
                                double a, double b, int n) {
    double h = (b - a) / n;
    double sum = 0.0;

    for (int i = 1; i < n; i++) {
        double x = a + i * h;
        sum += func.evaluate(x);
    }

    return h / 2 * (func.evaluate(a) + func.evaluate(b) + 2 * sum);
}
```



3.5 Метод Симпсона

```
private static double simpson(FunctionEvaluator func,
                              double a, double b, int n) {
    double h = (b - a) / n;
    double sum = func.evaluate(a) + func.evaluate(b);

    for (int i = 1; i < n; i++) {
        double x = a + i * h;
        sum += (i % 2 == 0) ? 2 * func.evaluate(x) : 4 * func.evaluate(x);
    }

    return sum * h / 3;
}
```