



УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Вычислительная математика

Лабораторная работа №2

Метод трапеций

Преподаватель: Перл Ольга Вячеславовна

Выполнили: Кульбако Артемий Юрьевич Р3212

## Описание метода

Метод трапеций – модификация метода прямоугольников, дающая более точные результаты. Идея заключается в разбиении площади под графиком подынтегральной функции на равные по ширине трапеции, и суммировании их площадей.

$$S_{\text{общ}} = S_1 + S_2 + \dots + S_n = \frac{y_0 + y_1}{2} h_1 + \frac{y_1 + y_2}{2} h_2 + \dots + \frac{y_{n-1} + y_n}{2} h_n = \\ = \int_a^b f(x) dx = \frac{1}{2} \sum_{i=1}^n h_i (y_{i-1} + y_i) = I_n$$

$$\text{где } y_0 = f(a), \quad y_n = f(b), \quad y_i = f(x_i), \quad h_i = x_i - x_{i-1}$$

После вычисления  $I_n$  проводится повторное интегрирование для  $I_{2n} = \frac{1}{2} \sum_{i=1}^{2n} h_i (y_{i-1} + y_i)$  и вычисляется погрешность по правилу Рунге:

$$\Delta_{2n} = \frac{|I_{2n} - I_n|}{3}$$

Если  $\Delta_{2n} < \varepsilon$  ( $\varepsilon$  – требуемая точность), то количество разбиений увеличивается в 2 раза, и  $\Delta_{2n}$  оценивается ещё раз.

## Вывод

Все три метода: прямоугольников, трапеций, парабол (Симпсона) являются модификациями метода Ньютона-Котеса, основанного на замене подынтегральной функции интерполяционным многочленом Лангража

$$\int_a^b y dx = (b-a) \sum_{i=0}^n H_i y_i$$

, где точность решения растёт с увеличением степени интерполяционного выражения. Погрешность же для каждого из методов определяется формулами:

1. Для средних прямоугольников:  $|\Delta| \leq \max_{x \in [a,b]} |f''(x)| \cdot \frac{(b-a)^3}{24 \cdot 2^2}, k = 2$
2. Для трапеций:  $|\Delta| \leq \max_{x \in [a,b]} |f''(x)| \cdot \frac{(b-a)^3}{12n^2}, k = 2$
3. Для парабол:  $|\Delta| \leq \max_{x \in [a,b]} |f''(x)| \cdot \frac{(b-a)^5}{189n^4}, k = 4$

где  $k$  – порядок точности

Это говорит нам о том, что метод трапеций менее точные чем метод парабол, при равном количестве разбиений. На практике же применяются метод оценки погрешности Рунге:

$$I - I_{\frac{h}{2}} \approx \frac{I_h - I}{2^k - 1}$$

## Примеры

Введите номер желаемой функции:

- 0.  $x^2 dx$
- 1.  $1/\ln(x) dx$
- 2.  $\cos(x)/(x+2) dx$
- 3.  $\sqrt{1 + 2x^2 - x^3} dx$
- 0

Введите пределы интегрирования через пробел:

0 2

Введите точность:

0.01

Значение интеграла = 2.671875

Количество разбиений = 16

Погрешность = 0.005208333333333333

Введите номер желаемой функции:

- 0.  $x^2 dx$
- 1.  $1/\ln(x) dx$
- 2.  $\cos(x)/(x+2) dx$
- 3.  $\sqrt{1 + 2x^2 - x^3} dx$
- 4

Ошибка ввода: введите целое число в [1 ; 5].

3

Введите пределы интегрирования через пробел:

1.2 2

Введите точность:

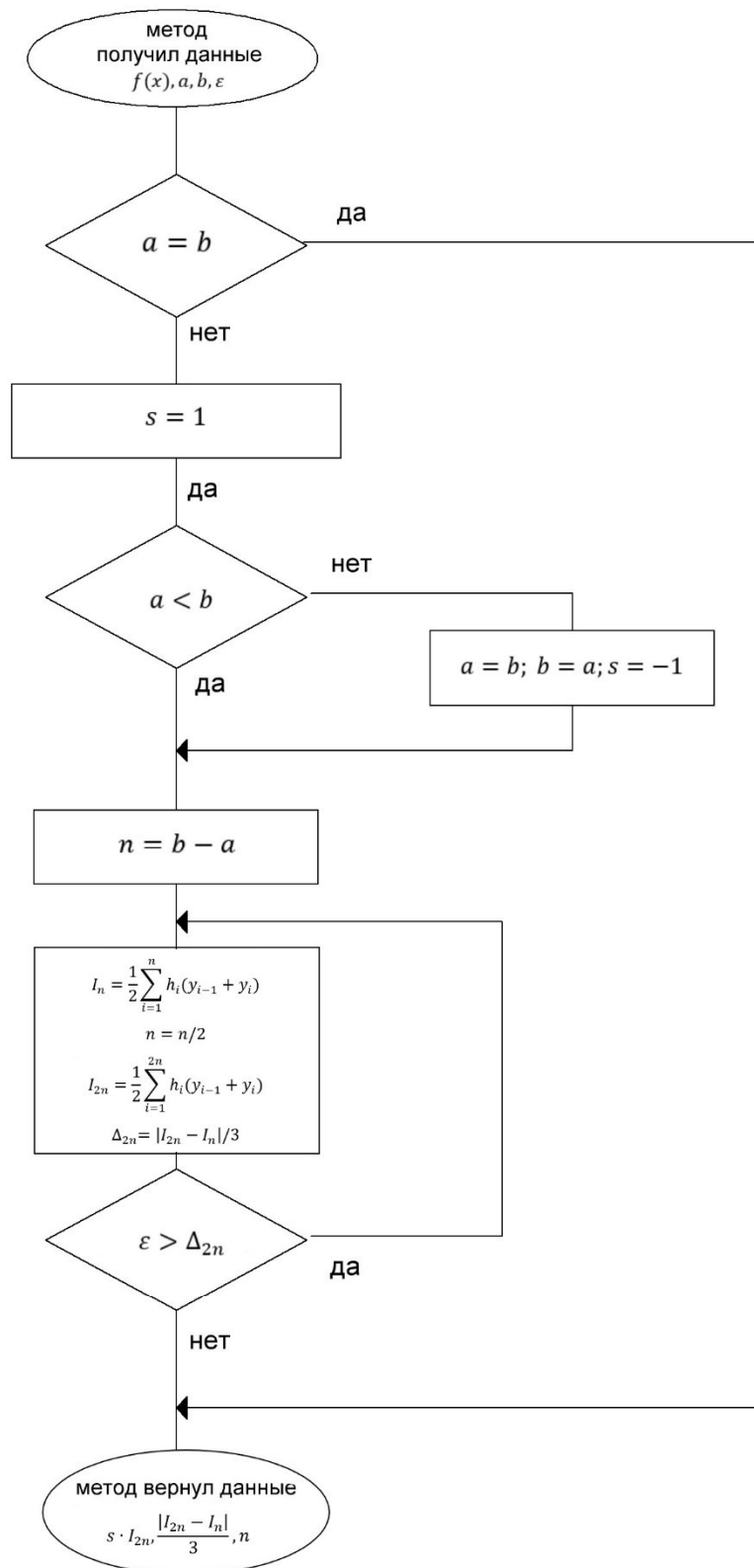
0.000001

Значение интеграла = 1.090122705480749

Количество разбиений = 512

Погрешность = 4.4018402671023676E-7

## Блок-схема



```

1 import kotlin.math.abs
2
3 /**
4  * Содержит результат решения интеграла.
5  * @property resValue значение интеграла.
6  * @property infelicity погрешность вычислений.
7  * @property blocks количество разбиений.
8  * @version 1.0
9  */
10 data class IntegralAnswer(val resValue: Double, val infelicity: Double, val blocks: Int)
11
12 /**
13  * Содержит результат пределы интегрирования.
14  * @param limits верхний и нижний предел (не упорядоченные).
15  * @property low нижний предел.
16  * @property high верхний предел.
17  * @property isSwitchedRange менялись ли пределы местами.
18  * @version 1.0
19  */
20 data class Limits(val limits: Pair<Double, Double>) {
21
22     val low: Double
23     val high: Double
24     val isSwitchedRange: Boolean
25
26     init {
27         if (limits.first > limits.second) {
28             low = limits.second
29             high = limits.first
30             isSwitchedRange = true
31         } else {
32             low = limits.first
33             high = limits.second
34             isSwitchedRange = false
35         }
36     }
37 }
38
39 /**
40  * Находит численное значение интеграла.
41  * @author Артемий Кульбако.
42  * @version 1.0
43  */
44 class IntegralSolver {
45
46     companion object {
47         /**
48          * Находит значение интеграла методом трапеций.
49          * @param mathFunc интеграл, который нужно посчитать.
50          * @param limits пределы интеграла.
51          * @param precision точность вычислений.
52          * @return IntegralAnswer.
53          * @version 1.0
54          */
55         fun integrateByTrapezoid(mathFunc: MathFunction, limits: Limits, precision: Double):
IntegralAnswer {
56
57             fun approximate(mathFunc: MathFunction, limits: Limits, step: Double): Double {
58                 var area = 0.0
59                 for (i in 0 until ((limits.high - limits.low) / step).toInt()) {
60                     area += step * 0.5 * (mathFunc.func(limits.low + i * step) + (mathFunc.func(limits.
low + (i + 1) * step)))
61                     if (area.isNaN() || area.isInfinite()) throw Exception("Функция не определена на
заданном отрезке.")
62                 }
63                 return area
64             }
65             var step = if (limits.low.let { it.isNaN() || it.isInfinite() } || limits.high.let { it.
isNaN() || it.isInfinite() })
(limits.high + precision) - (limits.low + precision) else limits.high - limits.low
66             var error: Double
67             var integralN: Double
68             var integral2N = approximate(mathFunc, limits, step)
69             do {
70                 integralN = integral2N
71                 step /= 2
72                 integral2N = approximate(mathFunc, limits, step)
73                 error = calcError(integral2N, integralN)
74             } while (error > precision)
75             if (limits.isSwitchedRange) integral2N = - integral2N
76

```

```
77         return IntegralAnswer(integral2N, error, ((limits.high - limits.low) / step).toInt())
78     }
79
80     private fun calcError(integralN: Double, integral2N: Double) = abs(integral2N - integralN) / 3
81 }
82 }
```

```
1 "C:\Program Files\Amazon Corretto\jdk1.8.0_232\bin\java.exe" -javaagent:C:\Users\pugalol\AppData\Local\
JetBrains\Toolbox\apps\IDEA-U\ch-1\193.6911.18\lib\idea_rt.jar=53179:C:\Users\pugalol\AppData\Local\
JetBrains\Toolbox\apps\IDEA-U\ch-1\193.6911.18\bin -Dfile.encoding=UTF-8 -classpath "C:\Program Files\
Amazon Corretto\jdk1.8.0_232\jre\lib\charsets.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\
ext\access-bridge-64.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\clldrdata.jar;C:\
Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\dnssns.jar;C:\Program Files\Amazon Corretto\jdk1.8
.0_232\jre\lib\ext\jaccess.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\jfxrt.jar;C:\
Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\localedata.jar;C:\Program Files\Amazon Corretto\
jdk1.8.0_232\jre\lib\ext\nashorn.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\sunec.jar
;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\sunjce_provider.jar;C:\Program Files\Amazon
Corretto\jdk1.8.0_232\jre\lib\ext\sunmscapi.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\
ext\sunpkcs11.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\ext\zipfs.jar;C:\Program Files\
Amazon Corretto\jdk1.8.0_232\jre\lib\jce.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\
jfxswt.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\jsse.jar;C:\Program Files\Amazon
Corretto\jdk1.8.0_232\jre\lib\management-agent.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib
\resources.jar;C:\Program Files\Amazon Corretto\jdk1.8.0_232\jre\lib\rt.jar;V:\itmo\2 course\
computational mathematics\lab2-21.03.20\out\production\lab2-21.03.20;C:\Users\pugalol\AppData\Local\
JetBrains\Toolbox\apps\IDEA-U\ch-1\193.6911.18.plugins\Kotlin\kotlinc\lib\kotlin-stdlib.jar;C:\Users\
pugalol\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch-1\193.6911.18.plugins\Kotlin\kotlinc\lib\kotlin-
reflect.jar;C:\Users\pugalol\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch-1\193.6911.18.plugins\Kotlin
\kotlinc\lib\kotlin-test.jar;C:\Users\pugalol\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch-1\193.6911.
18.plugins\Kotlin\kotlinc\lib\kotlin-stdlib-jdk7.jar;C:\Users\pugalol\AppData\Local\JetBrains\Toolbox\
apps\IDEA-U\ch-1\193.6911.18.plugins\Kotlin\kotlinc\lib\kotlin-stdlib-jdk8.jar" ShellKt
2 Введите номер желаемой функции:
3 0.  $x^2 dx$ 
4 1.  $1/\ln(x) dx$ 
5 2.  $\cos(x)/(x+2) dx$ 
6 3.  $\sqrt{1 + 2x^2 - x^3} dx$ 
7 0
8 \Введите пределы интегрирования через пробел:
9 0 2
10 Введите точность:
11 0.0001
12 Значение интеграла = 2.666748046875
13 Количество разбиений = 128
14 Погрешность = 8.138020833333333E-5
15
16 Process finished with exit code 0
17
```