МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Нижегородский государственный технический университет

им. Р.Е. Алексеева» (НГТУ)

Кафедра: «Цифровая экономика»

Дисциплина: «Численные методы»

**Лабораторная работа №5**

**«Численное интегрирование»**

Выполнил:

студент 3-го курса группы 21-САИ

Краличев Игорь Евгеньевич

Проверил:

д.ф.м.н., проф. Катаева Лилия Юрьевна

14.01.2024  
Подпись преподавателя : \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород, 2023

Оглавление

[Постановка задач 2](#_Toc156271029)

[*Задание №1* 2](#_Toc156271030)

[*Задание №2* 2](#_Toc156271031)

[*Задание №3* 3](#_Toc156271032)

[*Задание №4* 3](#_Toc156271033)

[*Задание №5* 3](#_Toc156271034)

[Методы решений и инструменты 4](#_Toc156271035)

[*Инструменты* 4](#_Toc156271036)

[*Методы решений* 4](#_Toc156271037)

[*Задание 1* 4](#_Toc156271038)

[*Задание 2* 5](#_Toc156271039)

[*Задание 3* 6](#_Toc156271040)

[*Задание 4* 6](#_Toc156271041)

[*Задание 5* 7](#_Toc156271042)

[Таблица идентификаторов 7](#_Toc156271043)

[*Задание 1* 7](#_Toc156271044)

[*Задание 2* 8](#_Toc156271045)

[*Задание 3* 8](#_Toc156271046)

[*Задание 4* 9](#_Toc156271047)

[*Задание 5* 10](#_Toc156271048)

[Ручной счет 10](#_Toc156271049)

[*Задание 1* 10](#_Toc156271050)

[*Задание 2* 13](#_Toc156271051)

[*Задание 3* 15](#_Toc156271052)

[*Задание 4* 16](#_Toc156271053)

[*Задание 5* 18](#_Toc156271054)

[Проверка вычислений 20](#_Toc156271055)

[*Excel* 20](#_Toc156271056)

[*Задание 1* 20](#_Toc156271057)

[*Задание 2* 22](#_Toc156271058)

[*Задание 3* 24](#_Toc156271059)

[*Задание 4* 25](#_Toc156271060)

[*Задание 5* 26](#_Toc156271061)

[*MathCad* 27](#_Toc156271062)

[*Задание 1* 27](#_Toc156271063)

[*Задание 2* 28](#_Toc156271064)

[*Задание 3* 31](#_Toc156271065)

[*Задание 4* 32](#_Toc156271066)

[*Задание 5* 34](#_Toc156271067)

[Реализация на С++ 35](#_Toc156271068)

[Реализация Java 42](#_Toc156271069)

[Результат всех реализаций 49](#_Toc156271070)

[Список литературы 50](#_Toc156271071)

# **Постановка задач**

## ***Задание №1***

Вариант 10

1) Вычислить интеграл по формулам левых и правых прямоугольников при n = 10, оценивая точность с помощью сравнения полученных результатов.

2) Вычислить интеграл по формуле средних прямоугольников, используя для оценки точности двойной просчёт при n1 = 8; n2 = 10.

## ***Задание №2***

Вариант 10

1) Вычислить интеграл по формуле трапеции с тремя десятичными знаками.

2) Вычислить интеграл по формуле Симпсона, при n = 8; оценить погрешность результата, составив таблицу конечных разностей.

## ***Задание №3***

Вариант 10

1) Найти приближенное значение интеграла по формуле «трёх восьмых», используя для контроля точности вычислений двойной просчёт при n1 = 9 и n2 = 12.

## ***Задание №4***

Вариант 10

1) Применяя экстраполяцию по Ричардсону, вычислить интеграл -

по формуле трапеций при n1 = 3, n2 = 6 и найти его уточнённое значение: , .

2) Применяя экстраполяцию по Ричардсону, вычислить интеграл –

по формуле Симпсона при n1 = 2, n2 = 4 и найти его уточнённое значение:

## ***Задание №5***

Вариант 10

Вычислить интеграл по формуле Гаусса, применяя для оценки точности двойной пересчёт (при n1 = 4 и n2 = 5).

**Методы решений и инструменты**

***Инструменты***

При решении лабораторной работы были использованы следующие инструменты:

1. Mathcad 15 (версия: M050 [], разрядность: x64);
2. Eclipse IDE (версия: 2022-09 (4.25.0), разрядность: x64);
3. Visual Studio IDE (версия: 17.7.3, разрядность: x64).

Mathcad 15 — это программное обеспечение для математического моделирования и анализа технических и научных данных. Оно предоставляет удобную среду для создания и решения математических выражений и уравнений, а также выполнения численных и символьных вычислений.

Eclipse IDE — это интегрированная среда разработки (IDE), используемая в компьютерном программировании. Оно содержит базовое рабочее пространство и расширяемую систему подключаемых модулей для настройки среды. Это вторая по популярности среда IDE для разработки на Java, и до 2016 года она была самой популярной. Eclipse написан в основном на Java и его основное применение заключается в разработке приложений Java, но он также может быть использован для разработки приложений на других языках программирования с помощью плагинов.

Visual Studio IDE- это интегрированная среда разработки (IDE) от компании Microsoft, предназначенная для разработки программного обеспечения. В Visual Studio предоставляются различные инструменты и функциональные возможности, упрощающие процесс разработки, отладки и тестирования приложений.

Visual Studio поддерживает множество языков программирования, включая C++, C#, Visual Basic, F#, JavaScript и другие. Для разработки на C++ в Visual Studio используется компилятор Microsoft C++, который обеспечивает мощные возможности компиляции и оптимизации кода.

Visual Studio обеспечивает разработчиков на C++ всеми необходимыми инструментами для создания высококачественного программного обеспечения. Отличительной чертой Visual Studio является его обширная функциональность и поддержка различных платформ и технологий, что делает его одним из популярных выборов для разработки на C++.

***Методы решений***

*Задание 1*

1) Для вычислений по формулам левых и правых прямоугольников разбиваем отрезок интегрирования на n частей с шагом . Далее составляем таблицу значений подынтегральной функции в точках деления отрезка. Затем находим сумму значений подынтегральной функции в точках деления отрезка для левых(S1) и првых(S2) прямоугольников:

где

Затем найдём приближённые значения интеграла. По формуле левых и правых прямоугольников:

За окночательное значение примем полусумму найденных значений, округлив результат до тысячных по формуле:

2) Для вычесления интеграла воспользуемся формулой средних прямоугольников:

Вычисления выполняются дважды при n1 = 8 и n2 = 10 и соответственно при и . Результат записывается в таблицы. Далее находим приближенные значения интеграла:

За конечный результат берём значение I2, так как оно точнее I1, поэтому II2.

*Задание 2*

1) Для достижения заданой точности в три десятичных знака после запятой необохидмо определить n так, чтобы выполнялось равенство:

где

подынтегральная функция. Поэтому находим сначала , затем по формуле (10) находим , полученное значение округляем ло целого в большую сторону. После находим n из формулы (9), полученное значение n округляем ло целого в большую сторону.

Вычисление интеграла производим по формуле:

где .

2) Согласно условию n=8, поэтому . Вычислительная формула имеет вид:

где

Для оценки точности полученного результата найдём конечные разности функций до разностей четвёртого порядка. Находим , и сотаточный член по формуле:

Погрешность вычислений можно оценить из соотношения:

*Задание 3*

1) Воспользуемся формулой «трёх восьмых», выражающей интеграл через суммы значений подынтегральной функции:

где

число разбиений n должно быть кратным трём.

В нашем случае при n1 = 9, , при n2 = 12, . За конечный результат берём значение I2, так как оно точнее I1, поэтому II2.

*Задание 4*

1) Если n1 = 3, то ; если n2 = 6, то , . Используя формулу трапеций, получим:

при n1 = 3:

при n2 = 6:

Затем находим уточнённое значение интеграла по формуле:

для формулы трапеций m = 2.

2) Если n1 = 2, то ; если n2 = 4, то , . Используя формулу трапеций, получим:

при n1 = 2:

при n2 = 4:

Затем находим уточнённое значение интеграла по формуле:

для формулы Симпсона m = 4.

*Задание 5*

1) Формула Гаусса имеет следующий вид:

где

Значения берём из таблицы квадратурных коэффициентов Гаусса. Вычисления удобно располагать в таблице. Далее считаем сумму по формуле:

следовательно,

**Таблица идентификаторов**

***Задание 1***

Таблица 1

Таблица идентификаторов, переменных, используемых для решения задачи

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название переменной | Excel | Mathcad | C++ | Java | Комментарий |
| i |  | i | i | i | Итерации |
| a |  | a | a | a | Нижняя граница интеграла |
| b |  | b | b | b | Верхняя граница интеграла |
| y |  | y | y | y | Подынтегральная функция |
| n |  | n,n1,n2 | n,n1,n2 | n,n1,n2 | Количество шагов |
| Zadanie |  | Zadanie11,  Zadanie 12 | Zadanie 11,  Zadanie 12 | Zadanie 11,  Zadanie 12 | Функция решающая выбранное задание |
| h |  | h,h1,h2 | h,h1,h2 | h,h1,h2 | Шаг |
| SL |  | SL | SL | SL | Сумма для левых прямоугольников |
| SP |  | SP | SP | SP | Сумма для правых прямоугольников |
| I |  | I.I1I2 | I.I1.I2 | I.I1.I2 | Возвращение приближённого интеграла |

***Задание 2***

Таблица 2

Таблица идентификаторов, переменных, используемых для решения задачи

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название переменной | Excel | Mathcad | C++ | Java | Комментарий |
| i |  | i | i | i | Итерации |
| a |  | a | a | a | Нижняя граница интеграла |
| b |  | b | b | b | Верхняя граница интеграла |
| y |  | y | y | y | Подынтегральная функция |
| n |  | n | n | n | Количество шагов |
| Zadanie |  | Zadanie21,  Zadanie 22 | Zadanie 21,  Zadanie 22 | Zadanie 21,  Zadanie 22 | Функция решающая выбранное задание |
| h |  | h,h1,h2 | h,h1,h2 | h,h1,h2 | Шаг |
| S |  | S1,S2,S3 | S1,S2,S3 | S1,S2,S3 | Суммирование значений функции для каждого массива |
| d |  | d0,d1,d2,  d3,d4 | d0,d1,d2,  d3,d4 | d0,d1,d2,  d3,d4 | Массивы с погрешностями |
| dl |  | dl | dl | dl | Оценка погрешности |
| I |  | I | I | I | Возвращение приближённого интеграла |

***Задание 3***

Таблица 3

Таблица идентификаторов, переменных, используемых для решения задачи

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название переменной | Excel | Mathcad | C++ | Java | Комментарий |
| i |  | i | i | i | Итерации |
| a |  | a | a | a | Нижняя граница интеграла |
| b |  | b | b | b | Верхняя граница интеграла |
| y |  | y | y | y | Подынтегральная функция |
| n |  | n,n1,n2 | n,n1,n2 | n,n1,n2 | Количество шагов |
| Zadanie |  | Zadanie 3 | Zadanie 3 | Zadanie 3 | Функция решающая выбранное задание |
| h |  | h,h1,h2 | h,h1,h2 | h,h1,h2 | Шаг |
| S |  | S1,S2,S3 | S1,S2,S3 | S1,S2,S3 | Суммирование значений функции для каждого массива |
| SP |  | SP | SP | SP | Сумма для правых прямоугольников |
| I |  | I1,I2 | I1,I2 | I1,I2 | Возвращение приближённого интеграла |

***Задание 4***

Таблица 4

Таблица идентификаторов, переменных, используемых для решения задачи

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название переменной | Excel | Mathcad | C++ | Java | Комментарий |
| i |  | i | i | i | Итерации |
| a |  | a | a | a | Нижняя граница интеграла |
| b |  | b | b | b | Верхняя граница интеграла |
| k |  | k | k | k | Номер варианта |
| c |  | c | c | c | Просто переменная |
| y |  | y | y | y | Подынтегральная функция |
| n |  | n,n1,n2 | n,n1,n2 | n,n1,n2 | Количество шагов |
| Zadanie |  | Zadanie 41,  Zadanie 42 | Zadanie 41,  Zadanie 42 | Zadanie 41,  Zadanie 42 | Функция решающая выбранное задание |
| h |  | h,h1,h2 | h,h1,h2 | h,h1,h2 | Шаг |
| S |  | S1,S2,S3 | S1,S2,S3 | S1,S2,S3 | Суммирование значений функции для каждого массива |
| SP |  | SP | SP | SP | Сумма для правых прямоугольников |
| I |  | I1,I2,I3 | I1,I2,I3 | I1,I2,I3 | Возвращение приближённого интеграла |

***Задание 5***

Таблица 5

Таблица идентификаторов, переменных, используемых для решения задачи

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название переменной | Excel | Mathcad | C++ | Java | Комментарий |
| i |  | i | i | i | Итерации |
| a |  | a | a | a | Нижняя граница интеграла |
| b |  | b | b | b | Верхняя граница интеграла |
| k |  | k | k | k | Номер варианта |
| c |  | c | c | c | Просто переменная |
| y |  | y | y | y | Подынтегральная функция |
| t,C |  | t1,t2, C1,C2 | t1,t2, C1,C2 | t1,t2, C1,C2 | Квадратурные коэффициента Гаусса |
| n |  | n,n1,n2 | n,n1,n2 | n,n1,n2 | Количество шагов |
| RAB |  | Zadanie 5 | Zadanie 5 | Zadanie 5 | Функция решающая выбранное задание |
| h |  | h,h1,h2 | h,h1,h2 | h,h1,h2 | Шаг |
| S |  | S1,S2,S3 | S1,S2,S3 | S1,S2,S3 | Суммирование значений функции для каждого массива |
| SP |  | SP | SP | SP | Сумма для правых прямоугольников |
| I |  | I1,I2 | I1,I2 | I1,I2 | Возвращение приближённого интеграла |

# **Ручной счет**

## ***Задание 1***

1) Исходные данные:

Шаг итераций:

Значения подынтегральной функции в точках xi представлена в таблице 6.

Таблица 6

Таблица со значениями подынтегральной функции в точках деления отрезка

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 0.8 | 0.3273 |
| 1 | 0.96 | 0.3295 |
| 2 | 1.12 | 0.3334 |
| 3 | 1.28 | 0.3388 |
| 4 | 1.44 | 0.3454 |
| 5 | 1.6 | 0.3530 |
| 6 | 1.76 | 0.3615 |
| 7 | 1.92 | 0.3706 |
| 8 | 2.08 | 0.3802 |
| 9 | 2.24 | 0.3903 |
| 10 | 2.4 | 0.4008 |

Затем найдём сумму полученных значений подынтегральной функции для левых(S1) и правых(S2) прямоугольников:

Далее найдём приближённые значений интеграла. По формуле левых и правых прямоугольников:

За окончательное значение примем полусумму найденных значений, округлив результат до тысячных:

2) Исходные данные:

Шаги итераций:

Так для вычисления интеграла воспользуемся формулой средних прямоугольников:

Составим таблицу 7 и 8 для значений подынтегральной функции в точках деления отрезка при n1 = 8 и n2 = 10:

Таблица 7

Таблица со значениями подынтегральной функции в точках деления отрезка при n1 = 8

|  |  |  |  |
| --- | --- | --- | --- |
| i |  |  |  |
| 0 | 0.4 | 0.45 | 0.2451 |
| 1 | 0.5 | 0.55 | 0.2017 |
| 2 | 0.6 | 0.65 | 0.1499 |
| 3 | 0.7 | 0.75 | 0.0894 |
| 4 | 0.8 | 0.85 | 0.0206 |
| 5 | 0.9 | 0.95 | -0.0552 |
| 6 | 1.1 | 1.05 | -0.1354 |
| 7 | 1.2 | 1.15 | -0.2158 |

Таблица 8

Таблица со значениями подынтегральной функции в точках деления отрезка при n2 = 10

|  |  |  |  |
| --- | --- | --- | --- |
| i |  |  |  |
| 0 | 0.4 | 0.44 | 0.2490 |
| 1 | 0.48 | 0.52 | 0.2156 |
| 2 | 0.56 | 0.60 | 0.1769 |
| 3 | 0.64 | 0.68 | 0.1326 |
| 4 | 0.72 | 0.76 | 0.0829 |
| 5 | 0.8 | 0.84 | 0.0278 |
| 6 | 0.88 | 0.92 | -0.0319 |
| 7 | 0.96 | 1 | -0.0950 |
| 8 | 1.04 | 1.08 | -0.1598 |
| 9 | 1.12 | 1.16 | -0.2237 |

Далее найдём сумму полученных значений подынтегральной функции при n1 = 8 и n2 = 10:

После находим приближённые значения интеграла по формуле (27):

За конечный результат берём значение I1, так как оно точнее значения I2, поэтому

## ***Задание 2***

1) Исходные данные:

Для достижения заданой точности в три десятичных знака после запятой необходимо определить n так, чтобы выполнялось равенство:

где

подынтегральная функция. Поэтому находим сначала :

M2 = 3, и из формулы (28) выражаем и находим

Вычисление интеграла производим по формуле:

где .

Значения подынтегральной функции в точках представлены в таблице 9.

Таблица 9

Таблица с первыми и последними пятью значениями подынтегральной функции в точках деления отрезка при n = 28

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 0.6 | 0.762493 |
| 1 | 0.632143 | 0.74552 |
| 2 | 0.664286 | 0.728831 |
| 3 | 0.696429 | 0.712466 |
| 4 | 0.728571 | 0.696457 |
| 24 | 1.371429 | 0.458271 |
| 25 | 1.403571 | 0.44992 |
| 26 | 1.435714 | 0.441832 |
| 27 | 1.467857 | 0.433995 |
| 28 | 1.5 | 0.426401 |

Для формулы (30) находим значение :

Сам интеграл будет равен:

2) Исходные данные:

Находим .

Вычислительная формула интеграла при n = 8 имеет вид:

где .

Значения подынтегральной функции в точках представлена в таблице 10.

Таблица 10

Таблица со значениями подынтегральной функции в точках деления отрезка при n = 8

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 0.4 | 0.587958 |
| 1 | 0.45 | 0.602542 |
| 2 | 0.5 | 0.621064 |
| 3 | 0.55 | 0.644737 |
| 4 | 0.6 | 0.675323 |
| 5 | 0.65 | 0.7155 |
| 6 | 0.7 | 0.769534 |
| 7 | 0.75 | 0.844674 |
| 8 | 0.8 | 0.954331 |

Затем находим значения интеграла по формуле (31):

Для оценки точности полученного результата составим таблицу 11 конечных разностей функций до разностей четвёртого порядка.

Таблица 11

Таблица со значениями подынтегральной функции в точках деления отрезка при n = 8

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i |  |  |  |  |  |
| 0 | 0.546302 | 4.154171 | -8.339385 | 12.01331 | -15.1492 |
| 1 | 4.700473 | -4.18521 | 3.673922 | -3.13589 | 2.547947 |
| 2 | 0.515259 | -0.51129 | 0.538034 | -0.58794 | 0.618325 |
| 3 | 0.003967 | 0.026742 | -0.049906 | 0.030385 | 0.057258 |
| 4 | 0.030708 | -0.02316 | -0.019521 | 0.087643 | -0.25087 |
| 5 | 0.007543 | -0.04269 | 0.068122 | -0.16323 |  |
| 6 | -0.03514 | 0.025436 | -0.095107 |  |  |
| 7 | -0.00971 | -0.06967 |  |  |  |
| 8 | -0.07938 |  |  |  |  |

Находим , и остаточный член по формуле:

## ***Задание 3***

Исходные данные:

Воспользуемся формулой «трёх восьмых», выражающей интеграл через суммы значений подынтегральной функции:

где , число разбиений n должно быть кратным трём.

Из исходных данных получаем при n1 = 9, .

Значение подынтегральных функций представлена в таблице 12.

Таблица 12

Таблица со значениями подынтегральной функции в точках деления отрезка при n1 = 9

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 1.1 | 0.902645 |
| 1 | 1.3 | 0.998602 |
| 2 | 1.5 | 1.10208 |
| 3 | 1.7 | 1.21151 |
| 4 | 1.9 | 1.325675 |
| 5 | 2.1 | 1.443634 |
| 6 | 2.3 | 1.564661 |
| 7 | 2.5 | 1.688191 |
| 8 | 2.7 | 1.813781 |
| 9 | 2.9 | 1.941085 |

И найдём значение интеграла при n1 = 9 по формуле (32):

В случае при n2 = 12, .

Значение подынтегральных функций представлена в таблице 13.

Таблица 13

Таблица со значениями подынтегральной функции в точках деления отрезка при n2 = 12

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 1.1 | 0.902645 |
| 1 | 1.25 | 0.973837 |
| 2 | 1.4 | 1.049508 |
| 3 | 1.55 | 1.128932 |
| 4 | 1.7 | 1.21151 |
| 5 | 1.85 | 1.296746 |
| 6 | 2 | 1.38423 |
| 7 | 2.15 | 1.473628 |
| 8 | 2.3 | 1.564661 |
| 9 | 2.45 | 1.6571 |
| 10 | 2.6 | 1.750752 |
| 11 | 2.75 | 1.845459 |
| 12 | 2.9 | 1.941085 |

И найдём значение интеграла при n2 = 12 по формуле (32):

Полученные значения совпадают с точностью до стотысячных, поэтому принимаем II2 .

## ***Задание 4***

1) Исходные данные:

,

Шаг итерации:

.

Значение подынтегральных функций представлена в таблице 14.

Таблица 14

Таблица со значениями подынтегральной функции в точках деления отрезка при n2 = 6

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 1 | 2 |
| 1 | 1.5 | 2.291288 |
| 2 | 2 | 2.645751 |
| 3 | 2.5 | 3.041381 |
| 4 | 3 | 3.464102 |
| 5 | 3.5 | 3.905125 |
| 6 | 4 | 4.358899 |

Воспользуемся формулой трапеций (30) для подсчёта интеграла при n1 = 3 и n2 = 6:

При n1 = 3:

При n2 = 6:

Затем находим уточнённое значение интеграла по формуле. где для формулы трапеций m = 2:

2) Исходные данные:

*,* найдём

Значения подынтегральных функций представлены в таблице 15.

Таблица 15

Таблица со значениями подынтегральной функции в точках деления отрезка при n2 = 4

|  |  |  |
| --- | --- | --- |
| i |  |  |
| 0 | 2 | 0.778151 |
| 1 | 3 | 1.041393 |
| 2 | 4 | 1.255273 |
| 3 | 5 | 1.431364 |
| 4 | 6 | 1.579784 |

По формуле Симпсона находим значения интеграла при n1 = 2 и n2 = 4:

При n1 = 2:

При n2 = 4:

Затем находим уточнённое значение интеграла по формуле (33), где для формулы Симпсона m = 4:

## ***Задание 5***

Исходные данные:

Формула Гаусса имеет следующий вид:

где

Значения подынтегральных функций при n1=4 и значения из таблицы квадратурных коэффициентов Гаусса представлены в таблице 16.

Таблица 16

Таблица со значениями подынтегральной функции в точках деления отрезка при n1 = 4

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i |  |  |  |  |  |
| 1 | 0.3478548 | -0.8611363 | -0.26114 | 0.714891 | 0.248678 |
| 2 | 0.6521451 | -0.339981 | 0.260019 | 1.219469 | 0.795271 |
| 3 | 0.6521451 | 0.339981 | 0.939981 | 1.413537 | 0.921831 |
| 4 | 0.3478548 | 0.8611363 | 1.461136 | 1.390025 | 0.483527 |

Найдём . Значения интеграла найдём по формуле (34):

Значения подынтегральных функций при n2=5 и значения из таблицы квадратурных коэффициентов Гаусса представлены в таблице 16.

Таблица 17

Таблица со значениями подынтегральной функции в точках деления отрезка при n2 = 5

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i |  |  |  |  |  |
| 1 | 0.2369269 | -0.9061798 | -0.30618 | 0.66342 | 0.157182 |
| 2 | 0.4786287 | -0.5384693 | 0.061531 | 1.059527 | 0.50712 |
| 3 | 0.568888 | 0 | 0.6 | 1.371989 | 0.780508 |
| 4 | 0.4786287 | 0.5384693 | 1.138469 | 1.411258 | 0.675469 |
| 5 | 0.2369269 | 0.9061798 | 1.50618 | 1.386222 | 0.328433 |

Найдём . Значения интеграла найдём по формуле (34):

# **Проверка вычислений**

## ***Excel***

### *Задание 1*



Рисунок 1.- Функция для вычисления интеграла по формуле левых и правых, средних прямоугольников



Рисунок 2.- Функция для вычисления интеграла по формуле левых и правых, средних прямоугольников

### *Задание 2*



Рисунок 3.- Функция для вычисления интеграла по формуле трапеций и по Формуле Симпсона



Рисунок 4.- Функция для вычисления интеграла по формуле трапеций и по Формуле Симпсона

### *Задание 3*



Рисунок 5.- Функция для вычисления интеграла по формуле "трех восьмых"



Рисунок 6.- Функция для вычисления интеграла по формуле "трех восьмых" при h1 и h2

### *Задание 4*



Рисунок 7.- Функция для вычисления интеграла с экстраполяцией по Ричардсону по формуле трапеции и Симпсона



Рисунок 8.-Функция для вычисления интеграла с экстраполяцией по Ричардсону (вариант 1)

### *Задание 5*



Рисунок 9.- Функция для вычисления интеграла по формуле Гаусса



Рисунок 10.- Функция для вычисления интеграла по формуле Гаусса

## ***MathCad***

### *Задание 1*

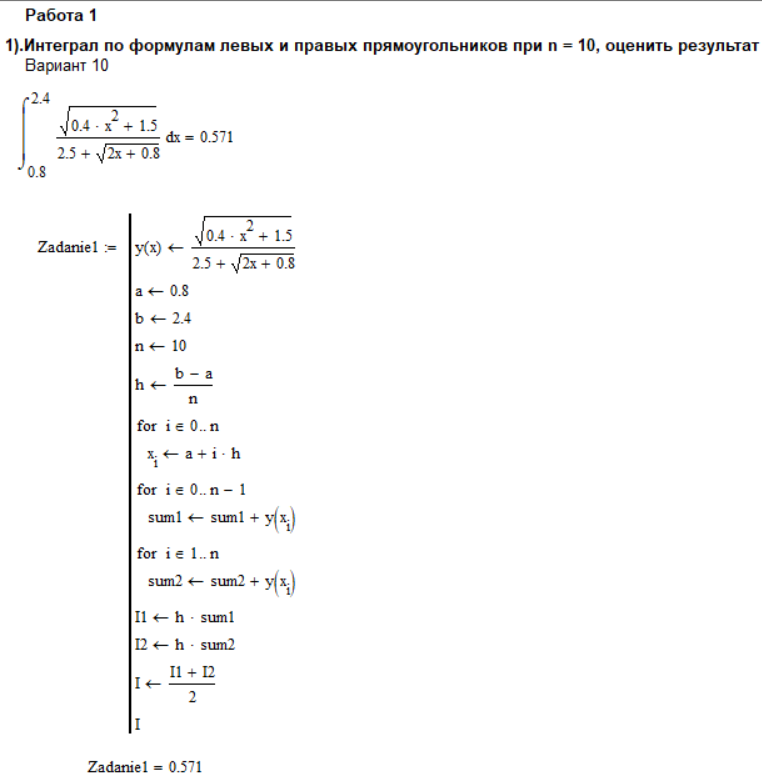


Рисунок 11.- Функция для вычисления интеграла по формуле левых и правых прямоугольников

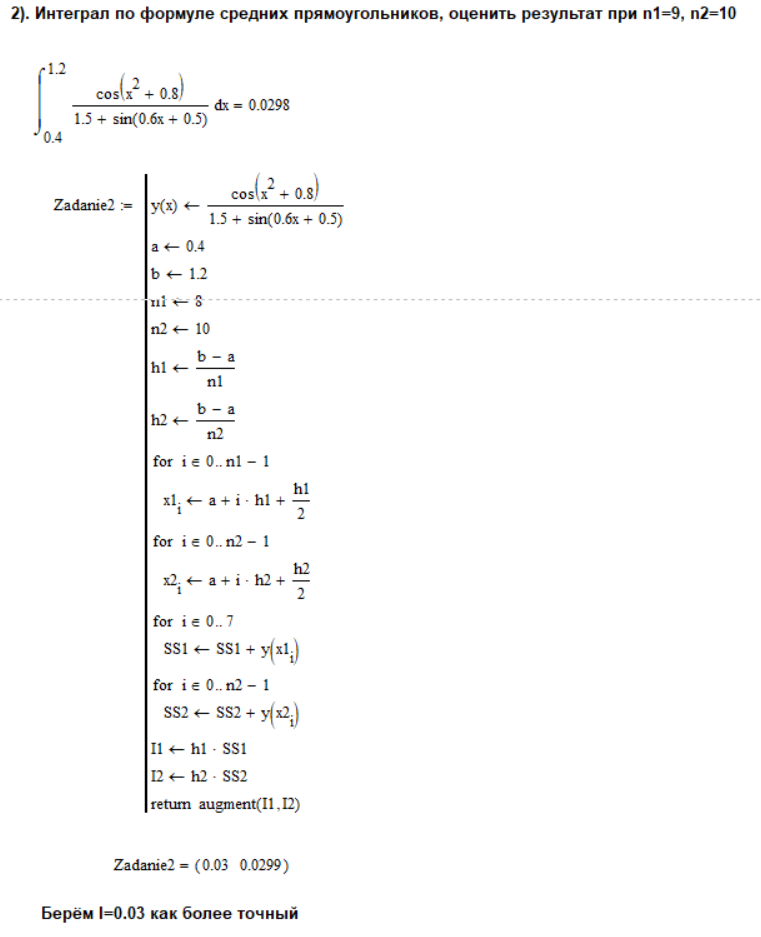


Рисунок 12.- Функция для вычисления интеграла по формуле средних прямоугольников

### *Задание 2*

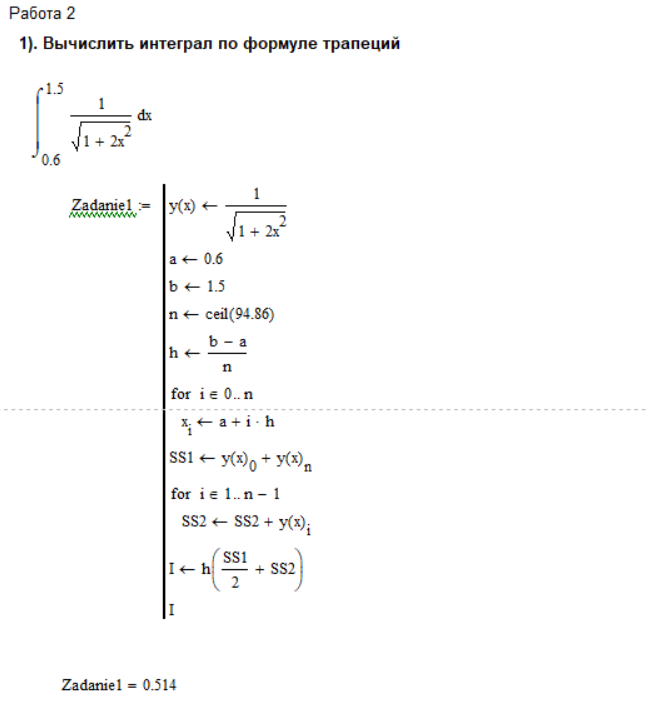


Рисунок 13.- Функция для вычисления интеграла по формуле трапеций

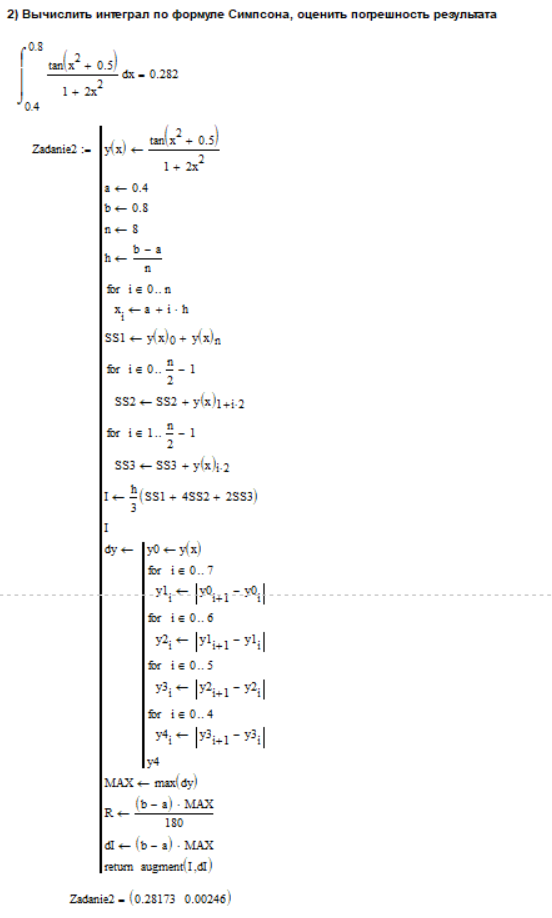


Рисунок 14.- Функция для вычисления интеграла по формуле Симпсона

### *Задание 3*

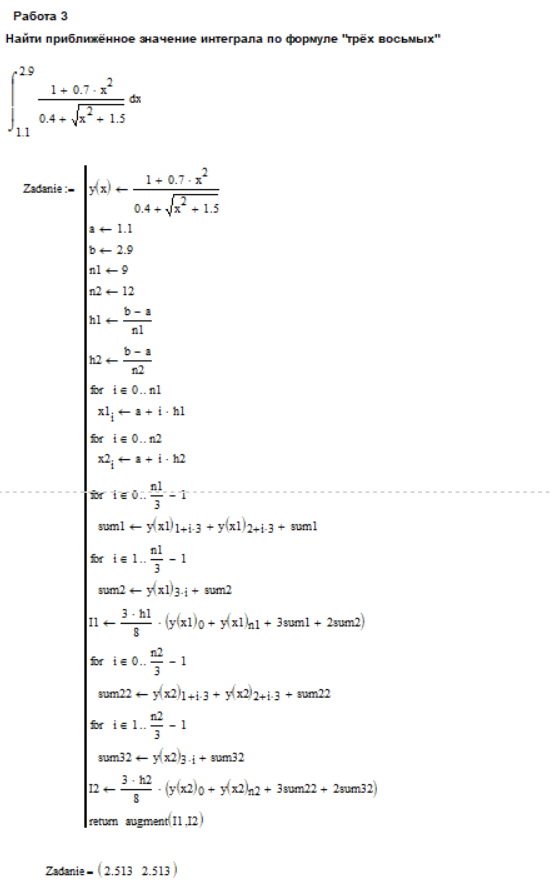


Рисунок 14.- Функция для вычисления интеграла по формуле "трех восьмых" при h1 и h2

### *Задание 4*

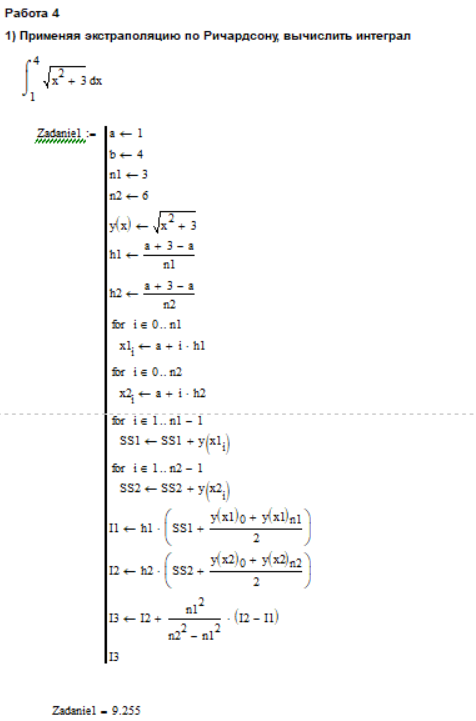


Рисунок 15.- Функция для вычисления интеграла с экстраполяцией по Ричардсону по формуле трапеций

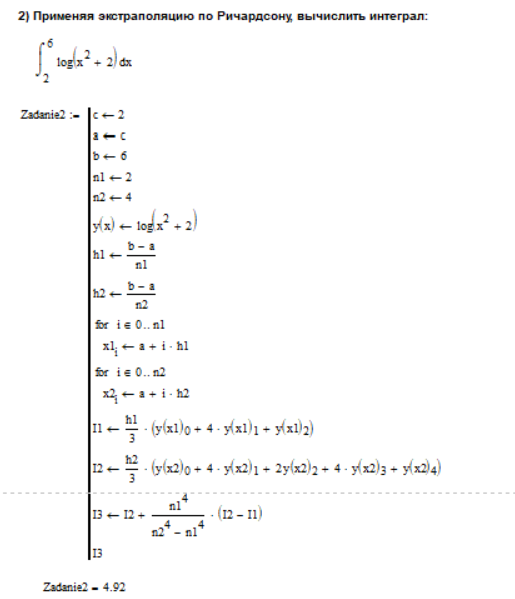


Рисунок 16.- Функция для вычисления интеграла с экстраполяцией по Ричардсону по формуле Симпсона

### *Задание 5*

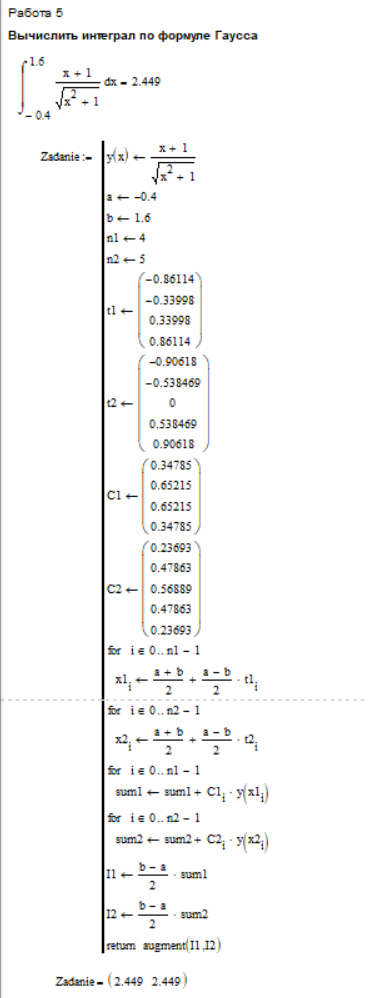


Рисунок 17.- Функция для вычисления интеграла по формуле Гаусса

Реализация на С++

Код программы для Численного дифференцирования и интегрирования на языке C++:

#include <iostream>

#include <math.h>

using namespace std;

// Вариант 10

double findH(double a, double b, int n) {

return (b - a) / n;

}

void Zadanie11() { // работа 1 задание 1

const int n = 10;

double a = 0.8;

double b = 2.4;

double h = findH(a, b, n);

double x[n + 1];

double y[n + 1];

for (int i = 0; i < n + 1; i++) {

x[i] = a + i \* h;

y[i] = (sqrt(0.4 \* pow(x[i], 2) + 1.5)) / (2.5 + sqrt(2.0 \* x[i] + 0.8));

}

double I1 = 0;

double I2 = 0;

for (int i = 0; i < n; i++) {

I1 += y[i];

I2 += y[i + 1];

}

I1 = I1 \* h;

I2 = I2 \* h;

cout << "По формуле левых прямоугольников: " << I1 << "\n";

cout << "По формуле правых прямоугольников: " << I2 << "\n";

cout << "Усреднённое значение интеграла: " << (I1 + I2) / 2 << "\n";

}

void Zadanie12() { // работа 1 задание 2

const int n1 = 8;

const int n2 = 10;

double a = 0.4;

double b = 1.2;

double h1 = findH(a, b, n1);

double h2 = findH(a, b, n2);

double x1[n1 + 1];

double sum1 = 0;

double x2[n2 + 1];

double sum2 = 0;

for (int i = 0; i < n1; i++) {

x1[i] = a + i \* h1 + h1 / 2;

cout << x1[i] << endl;

sum1 += (cos(pow(x1[i], 2) + 0.8)) / (1.5 + sin(0.6 \* x1[i] + 0.5));

//cout << sum1;

}

for (int i = 0; i < n2; i++) {

x2[i] = a + i \* h2 + h2 / 2;

sum2 += (cos(pow(x2[i], 2) + 0.8)) / (1.5 + sin(0.6 \* x2[i] + 0.5));

}

cout << "приближенное значение интеграла при n = 10: " << sum1 \* h1 << "\n";

cout << "приближенное значение интеграла при n = 8: " << sum2 \* h2 << "\n";

}

double diff(double x) {

return ((3 \* x \* x) / pow((x \* x + 1), 5.0 / 2.0)) - ((1) / (pow((x \* x + 1), 3.0 / 2.0)));

}

void Zadanie21() { // работа 2 задание 1

double a = 0.6;

double b = 1.5;

double maxi = max(abs(diff(a)), abs(diff(b)));

int n = int(sqrt((pow(1.5 - 0.6, 3) \* 3) / (12 \* 0.0005))) + 1; // n = 27

cout << n << endl;

double h = findH(a, b, n);

double\* x = new double[n + 1];

double sum1 = 0;

double sum2 = 0;

for (int i = 0; i < n + 1; i++) {

x[i] = a + i \* h;

sum2 += 1 / (sqrt(2.0 \* x[i] \* x[i] + 1));

}

sum1 += 1 / (sqrt(2.0 \* x[0] \* x[0] + 1));

sum1 += 1 / (sqrt(2.0 \* x[n] \* x[n] + 1));

sum2 -= sum1;

cout << "Приближённое значение: " << h \* ((sum1 / 2) + sum2) << "\n";

}

void Zadanie22() { // работа 2 задание 2

const int n = 8;

double a = 0.4;

double b = 0.8;

double h = findH(a, b, n);

double x[n + 1];

double y[n + 1];

for (int i = 0; i < n + 1; i++) {

x[i] = a + i \* h;

cout << tan(pow(x[i], 2) + 0.5) << endl;

y[i] = tan(x[i] \* x[i] + 0.5) / (1.0 + 2.0 \* pow(x[i], 2));

//cout << y[i] << endl;

}

double I = (h / 3) \* (y[0] + 4 \* y[1] + 2 \* y[2] + 4 \* y[3] + 2 \* y[4] + 4 \* y[5] + 2 \* y[6] + 4 \* y[7] + y[8]);

cout << "Приближённое значение: " << I << "\n";

double tr[n + 1][5];

for (int i = 0; i < n + 1; i++) {

tr[i][0] = y[i];

}

int k = 1;

for (int i = 8; i > 4; i--) {

for (int j = 0; j < i; j++) {

tr[j][k] = tr[j + 1][k - 1] - tr[j][k - 1];

}

k++;

}

double maxi = abs(tr[0][4]);

for (int i = 0; i < 5; i++) {

if (abs(tr[i][4]) > maxi) maxi = abs(tr[i][4]);

}

double R = ((b - a) \* maxi) / 180;

cout << "R = " << R << "\n";

double delI = (b - a) \* maxi;

//cout << "dI = " << delI << "\n";

if (delI < 0.00005) cout << "Полученные знаки верны!\n";

}

void Zadanie31() { // работа 3

const int n1 = 9;

const int n2 = 12;

double a = 1.1;

double b = 2.9;

double h1 = findH(a, b, n1);

double h2 = findH(a, b, n2);

double x1[n1 + 1];

double sum1 = 0, sum2 = 0, sum3 = 0;

for (int i = 0; i < n1 + 1; i++) {

x1[i] = a + i \* h1;

if (i % 3 == 0 && i > 0 && i < n1) { sum3 += (1 + 0.7 \* pow(x1[i], 2)) / (0.4 + sqrt(pow(x1[i], 2) + 1.5)); }

if (i % 3 != 0 && i > 0 && i < n1) {

sum2 += (1 + 0.7 \* pow(x1[i], 2)) / (0.4 + sqrt(pow(x1[i], 2) + 1.5));

}

if (i == 0 || i == n1) {

sum1 += (1 + 0.7 \* pow(x1[i], 2)) / (0.4 + sqrt(pow(x1[i], 2) + 1.5));

}

}

double I1 = h1 \* (3.0 / 8.0) \* (sum1 + 3 \* sum2 + 2 \* sum3);

double x2[n2 + 1];

sum1 = 0, sum2 = 0, sum3 = 0;

for (int i = 0; i < n2 + 1; i++) {

x2[i] = a + i \* h2;

if (i % 3 == 0 && i > 0 && i < n2) {

sum3 += (1 + 0.7 \* pow(x2[i], 2)) / (0.4 + sqrt(pow(x2[i], 2) + 1.5));

}

if (i % 3 != 0 && i > 0 && i < n2) {

sum2 += (1 + 0.7 \* pow(x2[i], 2)) / (0.4 + sqrt(pow(x2[i], 2) + 1.5));

}

if (i == 0 || i == n2) {

sum1 += (1 + 0.7 \* pow(x2[i], 2)) / (0.4 + sqrt(pow(x2[i], 2) + 1.5));

}

}

double I2 = h2 \* (3.0 / 8.0) \* (sum1 + 3 \* sum2 + 2 \* sum3);

cout << "Приближённое значение при n=9 I1 = " << I1 << "\n";

cout << "Приближённое значение при n=12 I2 = " << I2 << "\n";

}

void Zadanie41() { // Работа 4 задание 1

const int n1 = 3;

const int n2 = 6;

double a = 0.1 \* 10;

double b = a + 3;

double c = 4 - 0.1 \* 10;

double h1 = findH(a, b, n1);

double h2 = findH(a, b, n2);

double x[n2 + 1];

double y[n2 + 1];

double sum = 0;

for (int i = 0; i < n2 + 1; i++) {

x[i] = a + i \* h2;

y[i] = (sqrt(x[i] \* x[i] + c));

if (i > 0 && i < n2) sum += (sqrt(x[i] \* x[i] + c));

}

double I1 = h1 \* (((y[0] + y[n2]) / 2) + y[2] + y[4]);

double I2 = h2 \* (((y[0] + y[n2]) / 2) + sum);

cout << "По формуле трапеции\nI1 = " << I1 << "\n";

cout << "I2 = " << I2 << "\n";

cout << "Приближённое значение: \n";

cout << I2 + ((pow(n1, 2)) / (pow(n2, 2) - pow(n1, 2))) \* (I2 - I1) << "\n";

}

void Zadanie42() { //Работа 4 задание 2

const int n1 = 2;

const int n2 = 4;

double c = 3 - 0.1 \* 10;

double a = c;

double b = c + 4;

double h1 = findH(a, b, n1);

double h2 = findH(a, b, n2);

double x[n2 + 1];

double y[n2 + 1];

for (int i = 0; i < n2 + 1; i++) {

x[i] = a + i \* h2;

y[i] = log10(pow(x[i], 2) + 2);

}

double I1 = (h1 / 3.0) \* (y[0] + 4 \* y[2] + y[4]);

double I2 = (h2 / 3.0) \* (y[0] + 4 \* y[1] + 2 \* y[2] + 4 \* y[3] + y[4]);

cout << "По формуле Симпсона\nI1 = " << I1 << "\n";

cout << "I2 = " << I2 << "\n";

cout << "Приближённое значение: \n";

cout << I2 + ((pow(n1, 4)) / (pow(n2, 4) - pow(n1, 4))) \* (I2 - I1) << "\n";

}

void Zadanie51() { //Работа 5

const int n1 = 4;

const int n2 = 5;

double a = -0.4;

double b = 1.6;

double h1 = findH(a, b, n1);

double h2 = findH(a, b, n2);

double c4[n1] = { 0.34785, 0.65215, 0.65215, 0.34785 };

double t4[n1] = { -0.86114, -0.33998, 0.33998, 0.86114 };

double x1[n1];

double sum = 0;

for (int i = 0; i < n1; i++) {

x1[i] = ((b + a) / 2.0) + ((b - a) / 2.0) \* t4[i];

sum += c4[i] \* ((x1[i] + 1.0) / (sqrt(pow(x1[i], 2.0) + 1.0)));

cout << ((x1[i] + 1.0) / (sqrt(pow(x1[i], 2) + 1.0))) << endl;

}

double I1 = ((b - a) / 2) \* sum;

double c5[n2] = { 0.23693, 0.47863, 0.56889, 0.47863, 0.23693 };

double t5[n2] = { -0.90618, -0.538469, 0, 0.538469, 0.90618 };

double x2[n2];

sum = 0;

for (int i = 0; i < n2; i++) {

x2[i] = ((b + a) / 2) + ((b - a) / 2) \* t5[i];

sum += c5[i] \* ((x2[i] + 1) / (sqrt(pow(x2[i], 2.0) + 1.0)));

}

double I2 = ((b - a) / 2) \* sum;

cout << "Приближённое значение I1 = " << I1 << "\n";

cout << "Приближённое значение I2 = " << I2 << "\n";

}

int main() {

setlocale(LC\_ALL, "rus");

cout << "Введите номер работы(1,2,3,4,5)\n";

cout << "Для выхода введите любое число, не равное номеру задания\n";

int input;

cin >> input;

while (input == 1 || input == 2 || input == 3 || input == 4 || input == 5) {

switch (input) {

case 1:

cout << "вывод первого задания: \n";

Zadanie11();

cout << "вывод второго задания: \n";

Zadanie12();

break;

case 2:

cout << "вывод первого задания: \n";

Zadanie21();

cout << "вывод второго задания: \n";

Zadanie22();

break;

case 3:

cout << "вывод:\n";

Zadanie31();

break;

case 4:

cout << "вывод первого задания: \n";

Zadanie41();

cout << "вывод второго задания: \n";

Zadanie42();

break;

case 5:

cout << "вывод:\n";

Zadanie51();

break;

}

cout << "для выхода введите любое число, не равное номеру задания\n";

cin >> input;

}

cout << "Обратная связь:\n";

cout << "Студент гр. 21-САИ, Краличев Игорь Евгеньевич, ikralichev@list.ru\n";

system("pause");

}

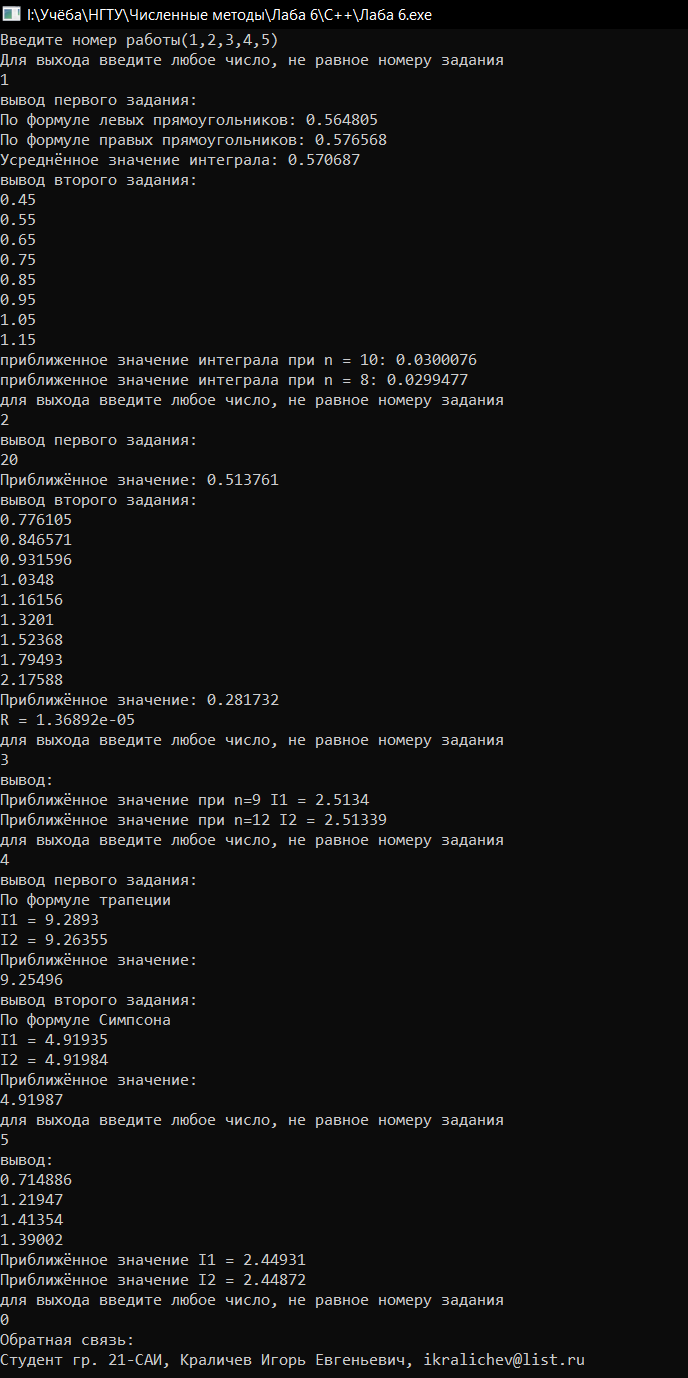


Рисунок 18.- Пример работы кода на С++

# **Реализация Java**

Код программы для Численного дифференцирования и интегрирования на языке Java:

package Lab6;

import java.lang.Math;

import java.util.Scanner;

public class Lab6 {

// Вариант 10

public static double findH(double a, double b, int n) {

return (b - a) / n;

}

public static void Zadanie11() { // работа 1 задание 1

final int n = 10;

double a = 0.8;

double b = 2.4;

double h = *findH*(a, b, n);

double[] x = new double[n + 1];

double[] y = new double[n + 1];

for (int i = 0; i < n + 1; i++) {

x[i] = a + i \* h;

y[i] = (Math.*sqrt*(0.4 \* Math.*pow*(x[i], 2) + 1.5)) / (2.5 + Math.*sqrt*(2.0 \* x[i] + 0.8));

}

double I1 = 0;

double I2 = 0;

for (int i = 0; i < n; i++) {

I1 += y[i];

I2 += y[i + 1];

}

I1 = I1 \* h;

I2 = I2 \* h;

System.***out***.println("По формуле левых прямоугольников: " + I1);

System.***out***.println("По формуле правых прямоугольников: " + I2);

System.***out***.println("Усреднённое значение интеграла: " + (I1 + I2) / 2);

}

public static void Zadanie12() { // работа 1 задание 2

final int n1 = 8;

final int n2 = 10;

double a = 0.4;

double b = 1.2;

double h1 = *findH*(a, b, n1);

double h2 = *findH*(a, b, n2);

double[] x1 = new double[n1 + 1];

double sum1 = 0;

double[] x2 = new double[n2 + 1];

double sum2 = 0;

for (int i = 0; i < n1; i++) {

x1[i] = a + i \* h1 + h1 / 2;

sum1 += (Math.*cos*(Math.*pow*(x1[i], 2) + 0.8)) / (1.5 + Math.*sin*(0.6 \* x1[i] + 0.5));

}

for (int i = 0; i < n2; i++) {

x2[i] = a + i \* h2 + h2 / 2;

sum2 += (Math.*cos*(Math.*pow*(x2[i], 2) + 0.8)) / (1.5 + Math.*sin*(0.6 \* x2[i] + 0.5));

}

System.***out***.println("приближенное значение интеграла при n = 10: " + sum1 \* h1);

System.***out***.println("приближенное значение интеграла при n = 8: " + sum2 \* h2);

}

public static double diff(double x) {

return ((3 \* x \* x) / Math.*pow*((x \* x + 1), 5.0 / 2.0)) - (1) / (Math.*pow*((x \* x + 1), 3.0 / 2.0));

}

public static void Zadanie21() { // работа 2 задание 1

double a = 0.6;

double b = 1.5;

double maxi = Math.*max*(Math.*abs*(*diff*(a)), Math.*abs*(*diff*(b)));

int n = (int) (Math.*sqrt*((Math.*pow*(1.5 - 0.6, 3) \* 3) / (12 \* 0.0005))) + 1; // n = 27

double h = *findH*(a, b, n);

double[] x = new double[n + 1];

double sum1 = 0;

double sum2 = 0;

for (int i = 0; i < n + 1; i++) {

x[i] = a + i \* h;

sum2 += 1 / (Math.*sqrt*(2.0 \* x[i] \* x[i] + 1));

}

sum1 += 1 / (Math.*sqrt*(2.0 \* x[0] \* x[0] + 1));

sum1 += 1 / (Math.*sqrt*(2.0 \* x[n] \* x[n] + 1));

sum2 -= sum1;

System.***out***.println("Приближённое значение: " + h \* ((sum1 / 2) + sum2));

}

public static void Zadanie22() { // работа 2 задание 2

final int n = 8;

double a = 0.4;

double b = 0.8;

double h = *findH*(a, b, n);

double[] x = new double[n + 1];

double[] y = new double[n + 1];

for (int i = 0; i < n + 1; i++) {

x[i] = a + i \* h;

y[i] = Math.*tan*(x[i] \* x[i] + 0.5) / (1.0 + 2.0 \* Math.*pow*(x[i], 2));

}

double I = (h / 3) \* (y[0] + 4 \* y[1] + 2 \* y[2] + 4 \* y[3] + 2 \* y[4] + 4 \* y[5] + 2 \* y[6] + 4 \* y[7] + y[8]);

System.***out***.println("Приближённое значение: " + I);

double[][] tr = new double[n + 1][5];

for (int i = 0; i < n + 1; i++) {

tr[i][0] = y[i];

}

int k = 1;

for (int i = 8; i > 4; i--) {

for (int j = 0; j < i; j++) {

tr[j][k] = tr[j + 1][k - 1] - tr[j][k - 1];

}

k++;

}

double maxi = Math.*abs*(tr[0][4]);

for (int i = 0; i < 5; i++) {

if (Math.*abs*(tr[i][4]) > maxi) maxi = Math.*abs*(tr[i][4]);

}

double R = ((b - a) \* maxi) / 180;

System.***out***.println("R = " + R);

double delI = (b - a) \* maxi;

if (delI < 0.00005) System.***out***.println("Полученные знаки верны!");

}

public static void Zadanie31() {

final int n1 = 9;

final int n2 = 12;

double a = 1.1;

double b = 2.9;

double h1 = *findH*(a, b, n1);

double h2 = *findH*(a, b, n2);

double[] x1 = new double[n1 + 1];

double sum1 = 0, sum2 = 0, sum3 = 0;

for (int i = 0; i < n1 + 1; i++) {

x1[i] = a + i \* h1;

if (i % 3 == 0 && i > 0 && i < n1) {

sum3 += (1 + 0.7 \* Math.*pow*(x1[i], 2)) / (0.4 + Math.*sqrt*(Math.*pow*(x1[i], 2) + 1.5));

}

if (i % 3 != 0 && i > 0 && i < n1) {

sum2 += (1 + 0.7 \* Math.*pow*(x1[i], 2)) / (0.4 + Math.*sqrt*(Math.*pow*(x1[i], 2) + 1.5));

}

if (i == 0 || i == n1) {

sum1 += (1 + 0.7 \* Math.*pow*(x1[i], 2)) / (0.4 + Math.*sqrt*(Math.*pow*(x1[i], 2) + 1.5));

}

}

double I1 = h1 \* (3.0 / 8.0) \* (sum1 + 3 \* sum2 + 2 \* sum3);

double[] x2 = new double[n2 + 1];

sum1 = 0;

sum2 = 0;

sum3 = 0;

for (int i = 0; i < n2 + 1; i++) {

x2[i] = a + i \* h2;

if (i % 3 == 0 && i > 0 && i < n2) {

sum3 += (1 + 0.7 \* Math.*pow*(x2[i], 2)) / (0.4 + Math.*sqrt*(Math.*pow*(x2[i], 2) + 1.5));

}

if (i % 3 != 0 && i > 0 && i < n2) {

sum2 += (1 + 0.7 \* Math.*pow*(x2[i], 2)) / (0.4 + Math.*sqrt*(Math.*pow*(x2[i], 2) + 1.5));

}

if (i == 0 || i == n2) {

sum1 += (1 + 0.7 \* Math.*pow*(x2[i], 2)) / (0.4 + Math.*sqrt*(Math.*pow*(x2[i], 2) + 1.5));

}

}

double I2 = h2 \* (3.0 / 8.0) \* (sum1 + 3 \* sum2 + 2 \* sum3);

System.***out***.println("Приближённое значение при n=9 I1 = " + I1);

System.***out***.println("Приближённое значение при n=12 I2 = " + I2);

}

public static void Zadanie41() { // Работа 4 задание 1

final int n1 = 3;

final int n2 = 6;

double a = 0.1 \* 10;

double b = a + 3;

double c = 4 - 0.1 \* 10;

double h1 = *findH*(a, b, n1);

double h2 = *findH*(a, b, n2);

double[] x = new double[n2 + 1];

double[] y = new double[n2 + 1];

double sum = 0;

for (int i = 0; i < n2 + 1; i++) {

x[i] = a + i \* h2;

y[i] = (Math.*sqrt*(x[i] \* x[i] + c));

if (i > 0 && i < n2) sum += (Math.*sqrt*(x[i] \* x[i] + c));

}

double I1 = h1 \* (((y[0] + y[n2]) / 2) + y[2] + y[4]);

double I2 = h2 \* (((y[0] + y[n2]) / 2) + sum);

System.***out***.println("По формуле трапеции\nI1 = " + I1);

System.***out***.println("I2 = " + I2);

System.***out***.println("Приближённое значение: ");

System.***out***.println(I2 + ((Math.*pow*(n1, 2)) / (Math.*pow*(n2, 2) - Math.*pow*(n1, 2))) \* (I2 - I1));

}

public static void Zadanie42() { //Работа 4 задание 2

final int n1 = 2;

final int n2 = 4;

double c = 3 - 0.1 \* 10;

double a = c;

double b = c + 4;

double h1 = *findH*(a, b, n1);

double h2 = *findH*(a, b, n2);

double[] x = new double[n2 + 1];

double[] y = new double[n2 + 1];

for (int i = 0; i < n2 + 1; i++) {

x[i] = a + i \* h2;

y[i] = Math.*log10*(Math.*pow*(x[i], 2) + 2);

}

double I1 = (h1 / 3.0) \* (y[0] + 4 \* y[2] + y[4]);

double I2 = (h2 / 3.0) \* (y[0] + 4 \* y[1] + 2 \* y[2] + 4 \* y[3] + y[4]);

System.***out***.println("По формуле Симпсона\nI1 = " + I1);

System.***out***.println("I2 = " + I2);

System.***out***.println("Приближённое значение: ");

System.***out***.println(I2 + ((Math.*pow*(n1, 4)) / (Math.*pow*(n2, 4) - Math.*pow*(n1, 4))) \* (I2 - I1));

}

public static void Zadanie51() { //Работа 5

final int n1 = 4;

final int n2 = 5;

double a = -0.4;

double b = 1.6;

double h1 = *findH*(a, b, n1);

double h2 = *findH*(a, b, n2);

double[] c4 = {0.34785, 0.65215, 0.65215, 0.34785};

double[] t4 = {-0.86114, -0.33998, 0.33998, 0.86114};

double[] x1 = new double[n1];

double sum = 0;

for (int i = 0; i < n1; i++) {

x1[i] = ((b + a) / 2.0) + ((b - a) / 2.0) \* t4[i];

sum += c4[i] \* ((x1[i] + 1.0) / (Math.*sqrt*(Math.*pow*(x1[i], 2.0) + 1.0)));

System.***out***.println((x1[i] + 1.0) / (Math.*sqrt*(Math.*pow*(x1[i], 2.0) + 1.0)));

}

double I1 = ((b - a) / 2) \* sum;

double[] c5 = {0.23693, 0.47863, 0.56889, 0.47863, 0.23693};

double[] t5 = {-0.90618, -0.538469, 0, 0.538469, 0.90618};

double[] x2 = new double[n2];

sum = 0;

for (int i = 0; i < n2; i++) {

x2[i] = ((b + a) / 2) + ((b - a) / 2) \* t5[i];

sum += c5[i] \* ((x2[i] + 1) / (Math.*sqrt*(Math.*pow*(x2[i], 2.0) + 1.0)));

}

double I2 = ((b - a) / 2) \* sum;

System.***out***.println("Приближённое значение I1 = " + I1);

System.***out***.println("Приближённое значение I2 = " + I2);

}

public static void main(String[] args) {

System.***out***.println("Введите номер работы(1,2,3,4,5)");

System.***out***.println("Для выхода введите любое число, не равное номеру задания");

Scanner scanner = new Scanner(System.***in***, "Cp1251");

int input = scanner.nextInt();

while (input == 1 || input == 2 || input == 3 || input == 4 || input == 5) {

switch (input) {

case 1:

System.***out***.println("вывод первого задания: ");

*Zadanie11*();

System.***out***.println("вывод второго задания: ");

*Zadanie12*();

break;

case 2:

System.***out***.println("вывод первого задания: ");

*Zadanie21*();

System.***out***.println("вывод второго задания: ");

*Zadanie22*();

break;

case 3:

System.***out***.println("вывод:");

*Zadanie31*();

break;

case 4:

System.***out***.println("вывод первого задания: ");

*Zadanie41*();

System.***out***.println("вывод второго задания: ");

*Zadanie42*();

break;

case 5:

System.***out***.println("вывод:");

*Zadanie51*();

break;

}

System.***out***.println("для выхода введите любое число, не равное номеру задания");

input = scanner.nextInt();

}

System.***out***.println("Обратная связь:");

System.***out***.println("Студент гр. 21-САИ, Краличев Игорь Евгеньевич, ikralichev@list.ru");

}

}

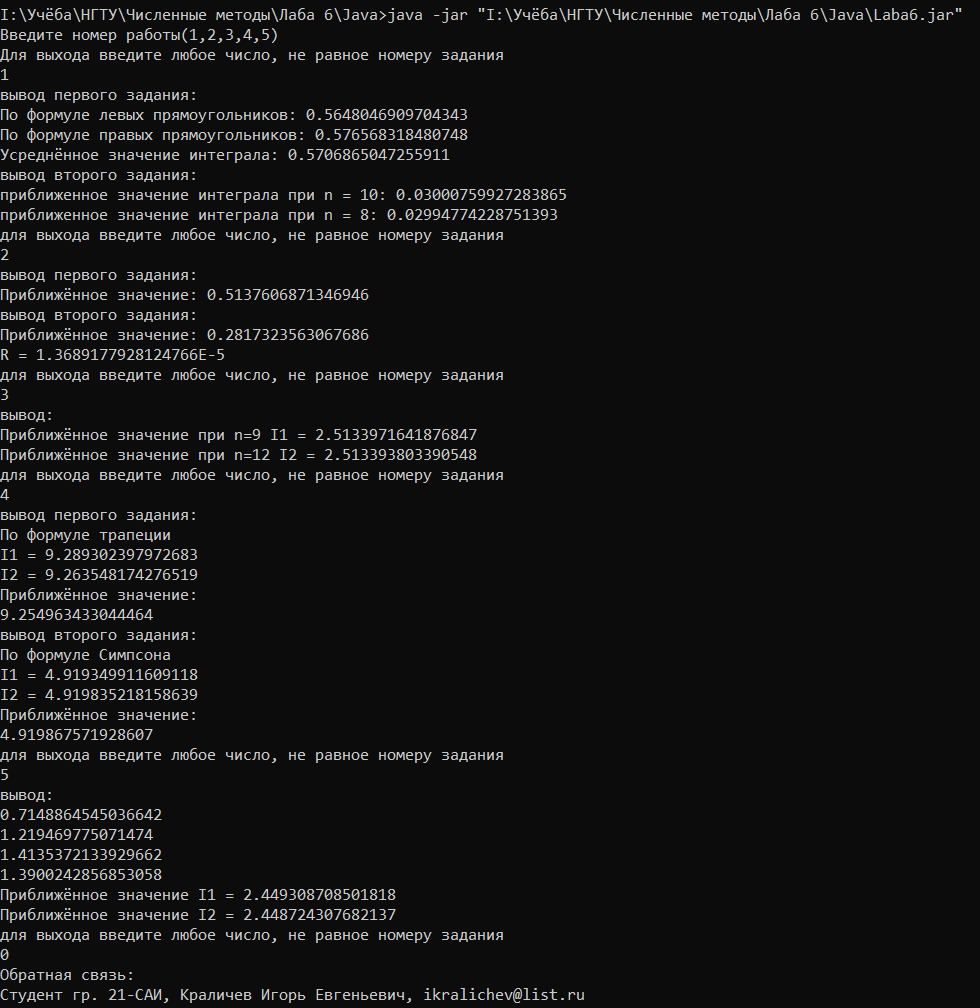
****

Рисунок 19.- Пример работы кода на Java

Результат всех реализаций

Таблица 6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Ручной счёт | Excel | Mathcad | C++ | Java |
| 1 | 2) 0.242698  0.247708 | 2) 0.222677  0.222694 | 2) 0.242698  0.247708 | 2) 0.242698  0.247708 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
| 5 |  |  |  |  |  |

Всеми инструментами можно достичь высокой точности, но среди прочих можно выделить Excel и Mathcad, которые подходят больше остальных для математических вычислений.

**Список литературы**

**1.** **Численное решение задач экономики с использованием EXCEL, C++ и MATLAB [Электронные текстовые данные] : Учеб.пособие / Л.Ю. Катаева [и др.]; НГТУ им.Р.Е.Алексеева. - Н.Новгород : [Изд-во НГТУ], 2020. - 230 с. : ил. - Прил.:c.188-230. - Библиогр.:с.187.:**

[**https://fdp.nntu.ru/books/Chisl\_reshenie\_zadach\_economiki/Chisl\_reshenie\_zadach\_economiki/assets/basic-html/index.html#189в**](https://fdp.nntu.ru/books/Chisl_reshenie_zadach_economiki/Chisl_reshenie_zadach_economiki/assets/basic-html/index.html#189%D0%B2)**.**

**2. Численные методы : Курс лекций / В.А. Срочко. - СПб.; М.; Краснодар : Лань, 2010. - 202 с. - (Учебники для вузов. Специальная литература). - Библиогр.:с.200. - ISBN 978-5-8114-1014-9 : 180-00.**

[**https://studfile.net/preview/5793014/page:4/**](https://studfile.net/preview/5793014/page:4/)

**3. Численные методы линейной алгебры : Учеб.пособие / Г.С. Шевцов, О.Г. Крюкова, Б.И. Мызникова. - 2-е изд.,испр.и доп. - СПб.; М.; Краснодар : Лань, 2011. - 495 с. - (Учебники для вузов. Специальная литература). - Предм.указ.:с.491-495. - Библиогр.:с.489-490. - ISBN 978-5-8114-1246-4 : 465-00.:**

[**https://dpm.pstu.ru/images/R/Z/shevcov\_lineynaya\_algebra.pdf**](https://dpm.pstu.ru/images/R/Z/shevcov_lineynaya_algebra.pdf)