Реферат

Данный документ является пояснительной запиской к курсовому проекту.

Тема данной дипломной работы — «Решение определенных интегралов методом Симпсона»,

Итог выполнения практической части работы является: приложение на языке высокого уровня с#.

Содержание

|  |  |
| --- | --- |
| Нормативные ссылки | 5 |
| Введение | 6 |
| Описание методов вычислительной математики, используемых при решении поставленной задачи | 7 |
| Описание среды разработки | 11 |
| Листинг | 12 |
| Описание программы | 13 |
| Результат машинного тестирования программы | 14 |
| Заключение | 16 |
| Список использованных источников | 17 |

Нормативные ссылки

ГОСТ 7.32-2017. Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления"

ГОСТ Р 7.0.5–2008 "Стандартизация в Российской Федерации. Основные положения"

Введение

Целью данной курсовой работы является написание консольного приложения на языке высокого уровня С#. Данное приложение решает определенные интегралы методом Симпсона.

Для достижения поставленной цели необходимо решить следующие взаимосвязанные задачи:

- Выбрать оптимальное средство разработки;

- Разработать приложение;

- Протестировать приложение и сравнить результаты с онлайн калькулятором.

2. Описание методов вычислительной математики, используемых при решении поставленной задачи.

Математическое обоснование вычисления определенного интеграла по методу парабол (Симпсона)

Если заменить график функции    на каждом отрезке    разбиения не отрезками прямых как в методах прямоугольников и трапеций, а дугами парабол, то получим более точную формулу приближенного вычисления определенного интеграла



.



Предварительно найдем площадь *S* криволинейной трапеции, ограниченной сверху графиком параболы   , сбоку – прямыми  ,   и снизу – отрезком    (см. рисунок 1).

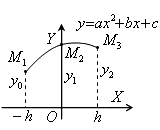


Рисунок 1 - Криволинейная фигура, ограниченная по сторонам прямыми    и   , а сверху графиком параболы



Пусть парабола проходит через три точки   ,   ,  , где  
 – ордината параболы в точке  ;  
  – ордината параболы в точке  ;  
  – ордината параболы в точке  .



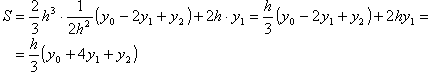
Площадь криволинейной фигуры, показанной на рисунке 3.3 равна



Выразим эту площадь через    . Из равенств для ординат    находим, что    ,  .



Подставляя эти значения **a** и **b** в равенство    , получаем



Получим теперь формулу парабол для вычисления интеграла   . Для этого отрезок [*a;b*] разобьем на *2n* равных частей (отрезков) длиной   c точками   , где   .



В точках деления    вычисляем значения подынтегральной функции   , где   .

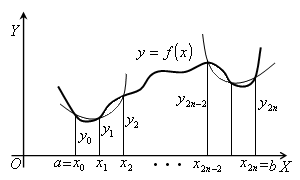
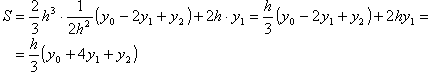


Рисунок 2 - Криволинейная фигура, полученная по методу Симпсона (парабол)

Заменяем каждую пару соседних элементарных криволинейных трапеций с основаниями, равным *h*, одной элементарной параболической трапецией с основанием, равным *2h*.

На отрезке    парабола проходит через три точки   ,   ,   . Используя формулу



находим  .



Аналогично находим , .



Сложив полученные равенства, имеем



или

.



Формула



называется формулой Симпсона (парабол).

Абсолютная погрешность вычисления по формуле оценивается соотношением   ,



где .



Отметим, что формула



дает точное значение интеграла    во всех случаях, когда *f(x)* – многочлен, степень которого меньше или равна трем (тогда   ).



Описание среды разработки

Используется среда разработки Visual studio 2019. Она является полнофункциональной интегрированной средой разработки для написания кода на языках программирования С++, C#, Python, Js, Visual Basic, его отладки, тестирования и развертывания кода на различных платформах (Windows, Linux, MacOs, Android), а также облачных и веб технологий (Azure, ASP.net, Node.js).

Листинг

using System;

namespace kursovoi\_project\_sharangiya

{

class Program

{

static void Main(string[] args)

{

// Нижний и верхний пределы интегрирования, количество шагов.

double a, b, steps;

Console.Write("Нижний предел интегрирования = ");

a = Convert.ToDouble(Console.ReadLine());

Console.Write("Верхний предел интегрирования = ");

b = Convert.ToDouble(Console.ReadLine());

Console.Write("Количество шагов = ");

steps = Convert.ToDouble(Console.ReadLine());

// Шаг интегрирования, сумма четных и нечетных элементов

double h, sumEven = 0, sumOdd = 0;

h = (b - a) / (2 \* steps);

for (int i = 1; i <= 2 \* steps - 1; i += 2)

{

// Значения с нечётными индексами, которые нужно умножить на 4.

sumOdd += function\_integral(a + h \* i);

// Значения с чётными индексами, которые нужно умножить на 2.

sumEven += function\_integral(a + h \* (i + 1));

}

Console.Write("Ответ:");

Console.Write((b - a) / (6 \* steps) \* (function\_integral(a) + 4 \* sumOdd + 2 \* sumEven - function\_integral(b)));

}

// Функция интеграла

static double function\_integral(double x)

{

return x \* x \* Math.Sqrt(4 - x \* x);

}

}

}

Описание программы

Блок схема программы (см. рисунок 3)

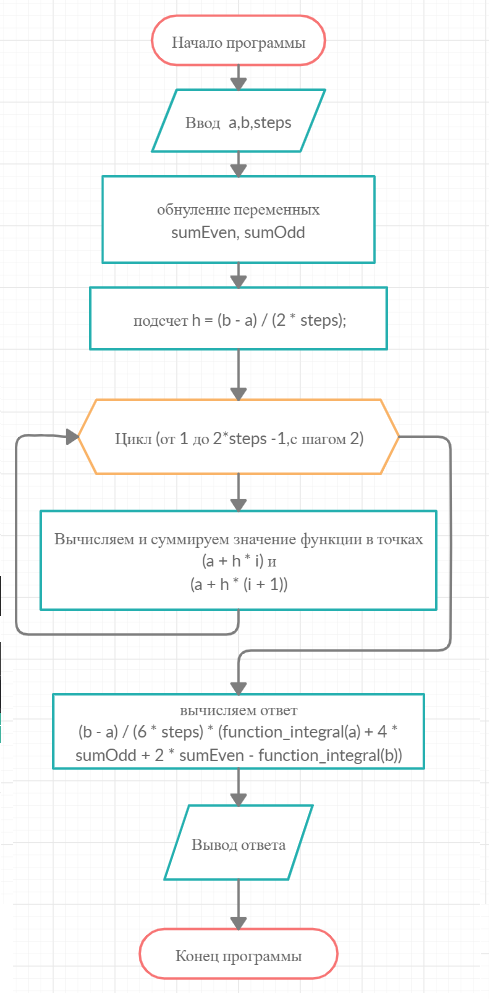


Рисунок 3 – Блок схема программы

Результат машинного тестирования программы

Вывод консольного приложения (см. рисунок 4)

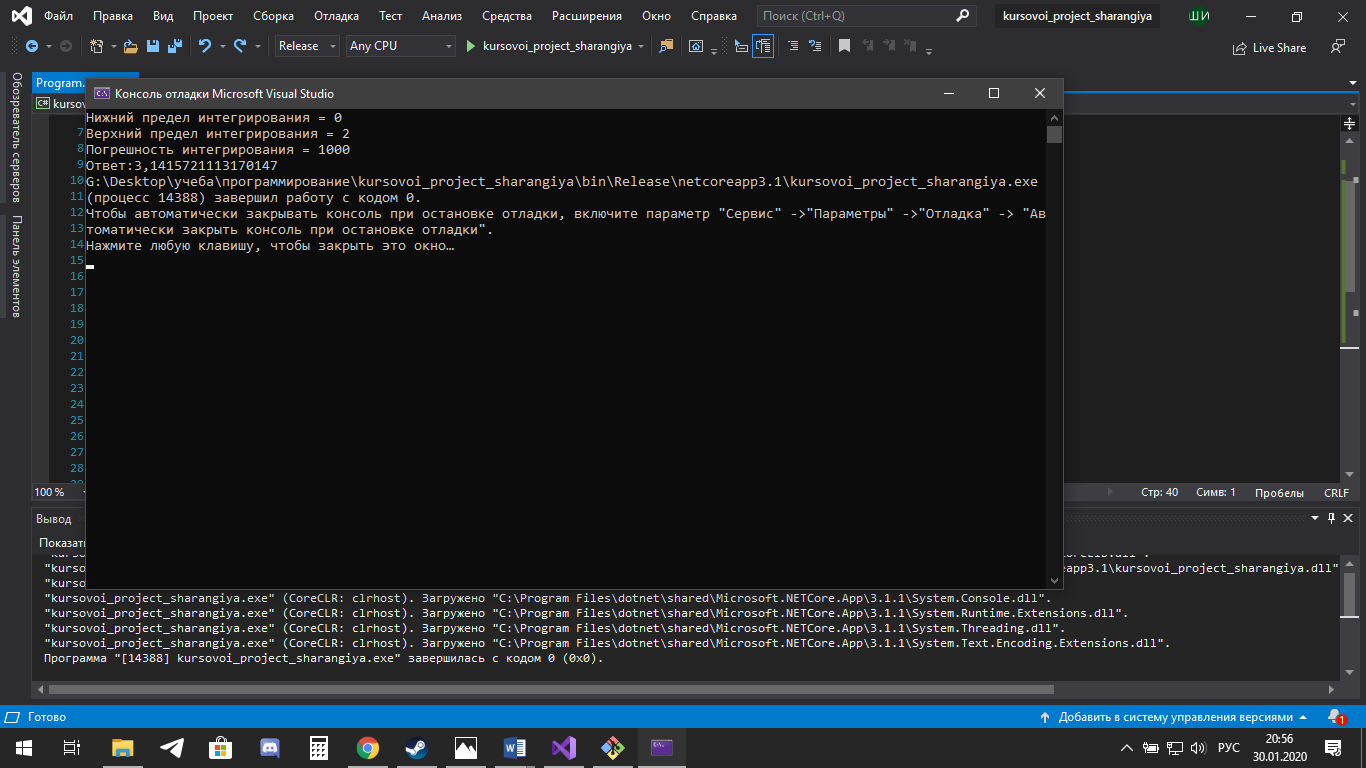


Рисунок 4 – Вывод выполнения программы.

Сравнение с онлайн калькулятором [электронный ресурс] [https://www.kontrolnaya-rabota.ru/s/integral/opredelennyij/?top=2&function=x%5E2\*sqrt%284-x%5E2%29&X=x&bottom=0](https://www.kontrolnaya-rabota.ru/s/integral/opredelennyij/?top=2&function=x%5E2*sqrt%284-x%5E2%29&X=x&bottom=0) (дата обращения:  30.01.2020) (см. рисунок 5-6)

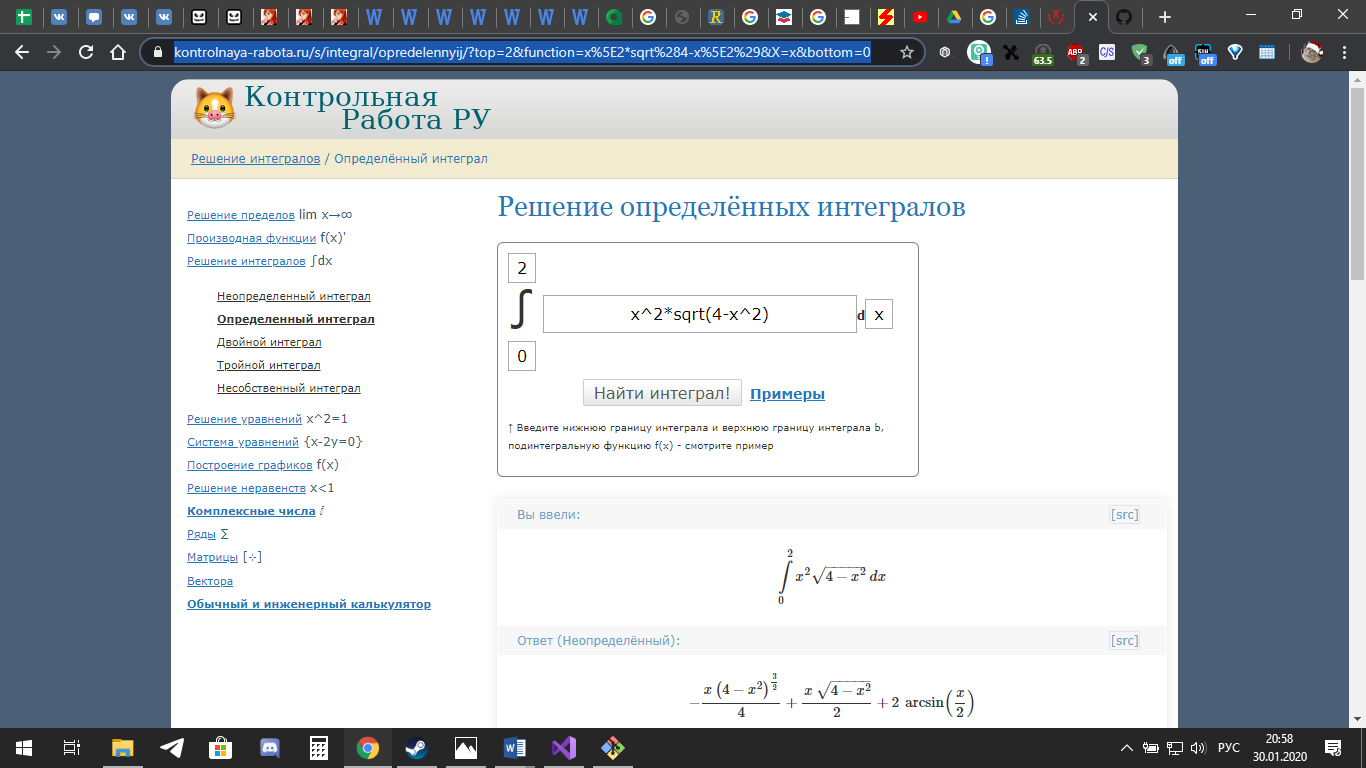


Рисунок 5 – Онлайн калькулятор

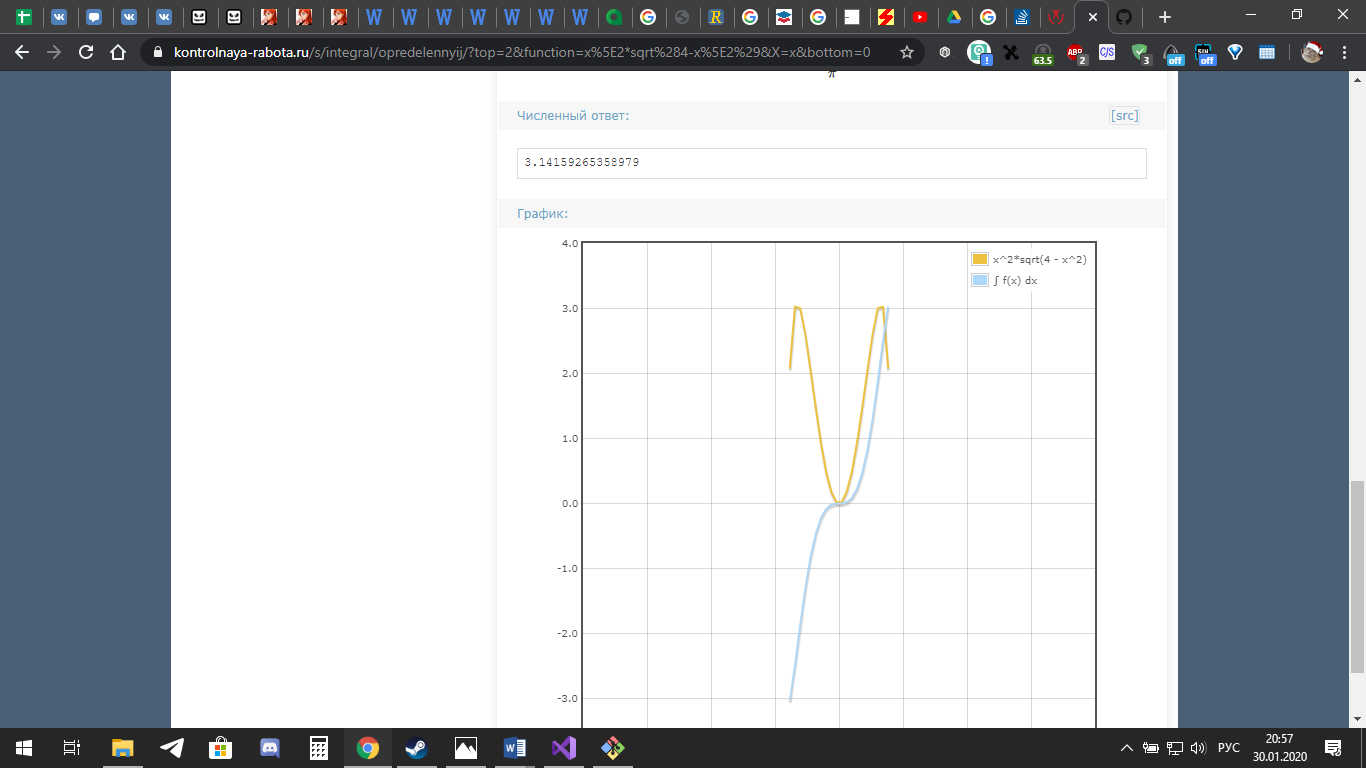


Рисунок 6 – Ответ в онлайн калькуляторе

Заключение

Были достигнуты цели курсовой работы, написание консольного приложения на языке высокого уровня С#.

Список использованных источников

Документация по C# [Электронный ресурс] https://docs.microsoft.com/ru-ru/dotnet/csharp/ (дата обращения:  30.01.2020)

Джозеф Албахари, Бен Албахари С# 6.0. Карманный справочник Изд-во Вильямс, 2016 г

C# для начинающих Пахомов Б.И. Изд-во «БХВ-Петербург» 2014 г

Приложение

