**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**«Сибирский государственный университет науки и технологий   
имени академика М.Ф. Решетнева»**

Институт информатики и телекоммуникаций

Кафедра информатики и вычислительной техники

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5**

Вычислительная математика

|  |
| --- |
| Приближенное вычисление определенных интегралов |

Руководитель К.А.Кириллов

подпись, дата инициалы, фамилия

Обучающийся БПИ20-02, 201219047 Р.А.Сухачев

номер группы, зачетной книжки подпись, дата инициалы, фамилия

Красноярск 2022 г.

# ЦЕЛЬ РАБОТЫ

Изучить приближенное вычисление определенных интегралов.

# постановка задачи

Вариант 29:

, a = 0, b = 2. (1)

1. Вычислить точное значения определенного интеграла по формуле Ньютона-Лейбница.
2. Последовательно увеличивая значение *N*, вычислить приближенное значение интеграла по формуле трапеций

, (2)

а также по формуле Симпсона

(3)

с точностью *ε*=0,001, *tk*=*a*+*k* (*k*=0,1,…,*N*), =(*b*-*a*)/*N.* В случае формулы Симпсона *N* четно.

1. Сравнить число узлов *N*тр.+1, требуемое для достижения заданной точности при вычислении интеграла по формуле трапеций, с числом узлов *N*С*.*+1, требуемым для достижения заданной точности при вычислении интеграла по формуле Симпсона.

# ХОД РАБОТЫ

1. Формулу (1) преобразовали в:

По формуле (4) программно вычислили точное значение определенного интеграла функцией Ньютона-Лейбница:

f_Newton_Leibniz

Рисунок 1 – точное значение определенного интеграла

1. Для вычисления значений абсолютных погрешностей приближенного вычисления интеграла по формуле Трапеций были использованы формулы (1, 2, 4) и была запрограммирована функция Трапеций:

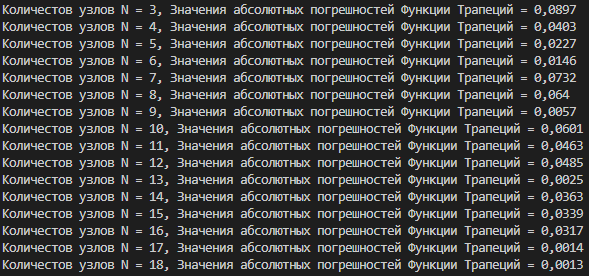


Рисунок 2 – значения абсолютных погрешностей приближенного вычисления интеграла по формуле Трапеций и количество узлов

Для вычисления значений абсолютных погрешностей приближенного вычисления интеграла по формуле Симпсона были использованы формулы (1, 3, 4) и была запрограммирована функция Симпсона:

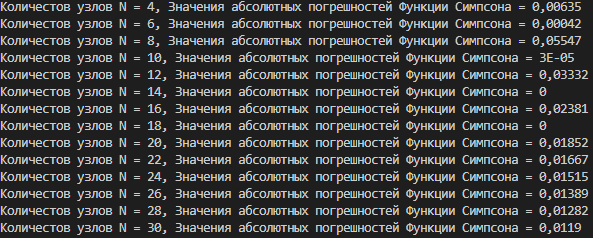


Рисунок 3 – значения абсолютных погрешностей приближенного вычисления интеграла по формуле Симпсона и количество узлов

1. Были вычислены результаты сравнения числа узлов Nтр.+1, требуемого для достижения заданной точности ε=0,001 при вычислении интеграла по формуле трапеций, с числом узлов NС.+1, требуемым для достижения заданной точности ε=0,001 при вычислении интеграла по формуле Симпсона:

countNodes

Рисунок 5 – результаты сравнения числа узлов

**Текст программы**

Содержимое файла Program.cs:

using System;

namespace lab5VAR29

{

    class Program

    {

        const double a = 0.0, b = 2.0, eps = 0.001;

        static double f(double x)

        {

            return x\*Math.Pow(2, -x);

        }

        static double F(double x)

        {

            return -((Math.Log(2)\*x+1)/(Math.Pow(Math.Log(2), 2)\*Math.Pow(2, x))); // -ln(2)\*x+1/ln^2(2)\*2^x

        }

        static double f\_Newton\_Leibniz()

        {

            return F(b)-F(a);

        }

        static double f\_Trapezium(out double N)

        {

            double res\_prev = 0.0, res\_now = 0.0;

            N = 2.0;

            do

            {

                double tau = (b-a)/N, temp = 0.0;

                for (double i = 0; i <= 2; i += tau)

                {

                    if (i == 0.0 || i == 2.0)

                        temp += (tau/2)\*(f(Math.Round(i, 5)));

                    else

                        temp += (tau/2)\*(2\*f(Math.Round(i, 5)));

                }

                res\_prev = res\_now;

                res\_now = temp;

                N++;

                Console.WriteLine($"Количестов узлов N = {N}, Значения абсолютных погрешностей Функции Трапеций = {Math.Round(Math.Abs(res\_now - f\_Newton\_Leibniz()), 4)}");

            } while(Math.Abs(res\_now - res\_prev) > eps);

            return res\_now;

        }

        static double f\_Simpson(out double N)

        {

            double res\_prev = 0.0, res\_now = 0.0;

            N = 2.0;

            do

            {

                double tau = (b-a)/N, temp = 0.0;

                int k = 0;

                for (double i = 0.0; i <= 2.0; i += tau)

                {

                    if (i == 0.0 || i == 2.0)

                        temp += (tau/3)\*(f(Math.Round(i, 5)));

                    else if ((k % 2) == 0 && i != 2.0 && i != 0.0)

                        temp += (tau/3)\*(2\*f(Math.Round(i, 5)));

                    else if ((k % 2) != 0 && i != 2.0 && i != 0.0)

                        temp += (tau/3)\*(4\*f(Math.Round(i, 5)));

                    k++;

                }

                res\_prev = res\_now;

                res\_now = temp;

                N += 2;

                Console.WriteLine($"Количестов узлов N = {N}, Значения абсолютных погрешностей Функции Симпсона = {Math.Round(Math.Abs(res\_now - f\_Newton\_Leibniz()), 5)}");

            } while(Math.Abs(res\_now - res\_prev) > eps);

            return res\_now;

        }

        static void Main(string[] args)

        {

            double Ntr, Nsim;

            Console.Clear();

            Console.OutputEncoding = System.Text.Encoding.UTF8;

            Console.WriteLine($"Функция Ньютона-Лейбница = {Math.Round(f\_Newton\_Leibniz(), 5)}");

            f\_Trapezium(out Ntr);

            Console.WriteLine();

            f\_Simpson(out Nsim);

            Console.WriteLine($"Количество узлов Функции Трапеций = {Ntr+1}\nКоличество узлов Функции Симпсона = {Nsim+1}\n{Ntr+1}>{Nsim+1}");

        }

    }

}

# ВЫВОДЫ

В ходе работы было изучено приближенное вычисление определенных интегралов.