**Молдавский государственный университет**

**Факультет математики и информатики**

**Департамент Информатики**

**КОМПЬЮТЕРНАЯ 2D ГРАФИКА**

**ЛАБОРАТОРНАЯ РАБОТА №1**

**Графические библиотеки. Базовые функции**

Выполнен студентом III курса

специальности Прикладная Информатика

Гуцу Даниил, подгруппа Безопасность

Проверил Геннадий Марин, lector universitar

**КИШИНЕВ – 2023**

ОГЛАВЛЕНИЕ

[1. УСЛОВИЯ ЛАБОРАТОРНОЙ РАБОТЫ 3](#_Toc145531864)

[2. ХОД ЛАБОРАТОРНОЙ РАБОТЫ 4](#_Toc145531865)

[2.1. Исходный код метода OnDraw() 5](#_Toc145531866)

[2.2. Результат отработки программы 7](#_Toc145531867)

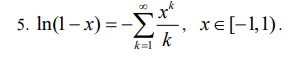
[ВЫВОД 10](#_Toc145531868)

# УСЛОВИЯ ЛАБОРАТОРНОЙ РАБОТЫ

1. Даны функция y=f(x) и её разложение в числовой ряд, например, . Составить на языке C++ собственную функцию (процедуру), зависящую от двух параметров и вычисляющую значение функции f для заданного значения аргумента x с указанной точностью ε.
2. Создать проект приложение типа Single Document в Microsoft Visual C++. Добавить в него код, рисующий отрезками прямых линий график функции f, указанным цветом на заданном интервале (a, b) с заданным шагом h. Для вычисления значений функции использовать собственную процедуру и указанную точность.
3. Добавить в проект код, рисующий другим цветом поверх первого графика поточечный график этой же функции, но вычисляющий её в процессе рисования при помощи стандартной библиотеки математических функции. Графики должны совпасть.
4. Добавить в проект код, создающий два текстовых файла с одинаковым количеством строк, совпадающим с количеством вычисляемых точек графика. Первый файл должен содержать для каждой точки: значение аргумента; значение функции, вычисленное при помощи собственной процедуры; значение функции, вычисленное при помощи библиотечных функции; абсолютную величину разности двух вычисленных значений функции; заданное значение точности. Этот файл использовать для визуального контроля того факта, что абсолютная величина разности в каждой строке не превосходит значение точности. Второй файл должен содержать для каждой строчки только значение аргумента и значение функции, вычисленное при помощи собственной процедуры. Числа в строках должны разделяться запятой или пробелом. Второй текстовый файл использовать в следующем пункте для дополнительного контроля.
5. Запустить специализированную программу построения графиков функций. Импортировать второй текстовый файл и построить график функции с использованием пар координат из текстового файла. Построить поверх полученного графика график этой же функции путём её аналитического задания. Графики должны совпасть. Провести анализ полученных результатов.

# ХОД ЛАБОРАТОРНОЙ РАБОТЫ

**Вариант №1.**



**Алгоритм:**

1. Инициализация. Задание начальных значений для интервала, задание значения для точности.
2. Рисование координатных осей.
3. Нахождение первых членов числового ряда, аналитическим путем:
   1. k1 = -(x1/1)
   2. k2 = -(x2/2)
   3. k3 = -(x3/3)
   4. …
4. Инициализация терма *t* значением 1, *summa* = -1, k = 0.
   1. Пока СДЕЛАЙ:
   3. *k/k*
   4. *k++*
   5. ЕСЛИ , ТО ВЕРНУТЬ *summa* ИНАЧЕ 4.1.
5. В основной программе вычисляются значения с помощью своей функции и табличной, записываются результаты в файл и выводятся рисунки двух графиков.

## Программа

Код класса MainDialog рисуещего графики:

namespace lab1;  
  
class MainDialog : Form  
{  
 private const string *fileOfMyFoo* = "C:\\Users\\Gutu\\RiderProjects\\Grafica\_lab1\\lab1\\FileOfMyFoo.txt";  
 private const string *fileOfAbsolute* = "C:\\Users\\Gutu\\RiderProjects\\Grafica\_lab1\\lab1\\FileToCheckAbsolute.txt";  
  
 public MainDialog()  
 {  
 Text = "Dialog based application";  
  
 StartPosition = FormStartPosition.*Manual*;  
 Location = new System.Drawing.Point(10, 10);  
 Size = new System.Drawing.Size(1300, 700);  
  
 Closed += new System.EventHandler(MainDialog\_Closed);  
  
 MouseDoubleClick +=  
 new MouseEventHandler(MyForm\_MouseDoubleClick);  
 }  
  
 private void MainDialog\_Closed(object sender, EventArgs e)  
 {  
 Dispose();  
 }  
  
  
 private void MyForm\_MouseDoubleClick(object sender, MouseEventArgs e)  
 {  
 mydraw();  
 }  
  
 private void mydraw()  
 {  
 Graphics g = Graphics.*FromHwnd*(this.Handle);  
  
 // Cream culorile  
 Color MYTGCOLOR = Color.*FromArgb*(255, 0, 0); //rosu  
 Color STTGCOLOR = Color.*FromArgb*(0, 255, 0); //verde  
  
 int cpx = (int)(this.ClientRectangle.Width / 2);  
 int cpy = (int)(this.ClientRectangle.Height / 2);  
  
 Pen AxPen = new Pen(Color.Blue, 1);  
 g.DrawLine(AxPen, 0, cpy, this.ClientRectangle.Width, cpy);  
 g.DrawLine(AxPen, cpx, 0, cpx, this.ClientRectangle.Height);  
  
  
 double a, b;  
  
 a = -1;  
 b = 1;  
  
 double precision = 0.005;  
  
  
 double scalex = this.ClientRectangle.Width / (b - a) \* 0.8;  
 double scaley = this.ClientRectangle.Height / 2.0 \* 0.8;  
  
  
 int graphsegments = 400;  
  
 double x1, x2, y1, y2, step;  
 int ix1, ix2, iy1, iy2;  
  
 step = (b - a) / (double)graphsegments;  
  
 x2 = a;  
 y2 = myFunc(x2, precision);  
  
 ix2 = cpx + (int)(x2 \* scalex);  
 iy2 = cpy - (int)(y2 \* scaley);  
  
 // Cream penite  
 Pen RedPen1 = new Pen(Color.Red, 1);  
 Pen StPen = new Pen(STTGCOLOR, 1);  
  
 // Cream pensule  
 SolidBrush GreenBrush = new SolidBrush(Color.Green);  
 SolidBrush MyBrush = new SolidBrush(MYTGCOLOR);  
 SolidBrush StBrush = new SolidBrush(STTGCOLOR);  
  
 while (x2 < b)  
 {  
 x1 = x2;  
 y1 = y2;  
 ix1 = ix2;  
 iy1 = iy2;  
 x2 += step;  
  
 y2 = myFunc(x2, precision);  
 ix2 = cpx + (int)(x2 \* scalex);  
 iy2 = cpy - (int)(y2 \* scaley);  
  
 g.DrawLine(RedPen1, ix1, iy1, ix2, iy2);  
 iy1 = cpy - (int)(Math.*Log*(1 - x1, Math.*E*) \* scaley);  
  
  
 g.DrawLine(StPen, ix1 + 0, iy1 - 2, ix1 + 0, iy1 + 2);  
 g.DrawLine(StPen, ix1 + 2, iy1 + 2, ix1 + 2, iy1 + 2);  
  
 string toFile = x1 + "\t" + y1 + "\t" + y2 + "\t" + Math.*Abs*(y1 - y2) + "\t" + precision;  
 MyService.*WriteLineToFile*(toFile, *fileOfAbsolute*);  
 }  
 }  
  
 double myFunc(double x, double eps)  
 {  
 int k = 1;  
 double sum = -1;  
 double t = 1;  
 do  
 {  
 sum += t;  
 t = Math.*Pow*(x, k) / k;  
 k++;  
 } while (Math.*Abs*(t) >= eps);  
  
 string toFile = x + "\t" + -sum;  
 MyService.*WriteLineToFile*(toFile, *fileOfMyFoo*);  
 return -sum;  
 }  
  
 protected override void OnPaint(PaintEventArgs e)  
 {  
 base.OnPaint(e);  
 mydraw();  
 }  
}

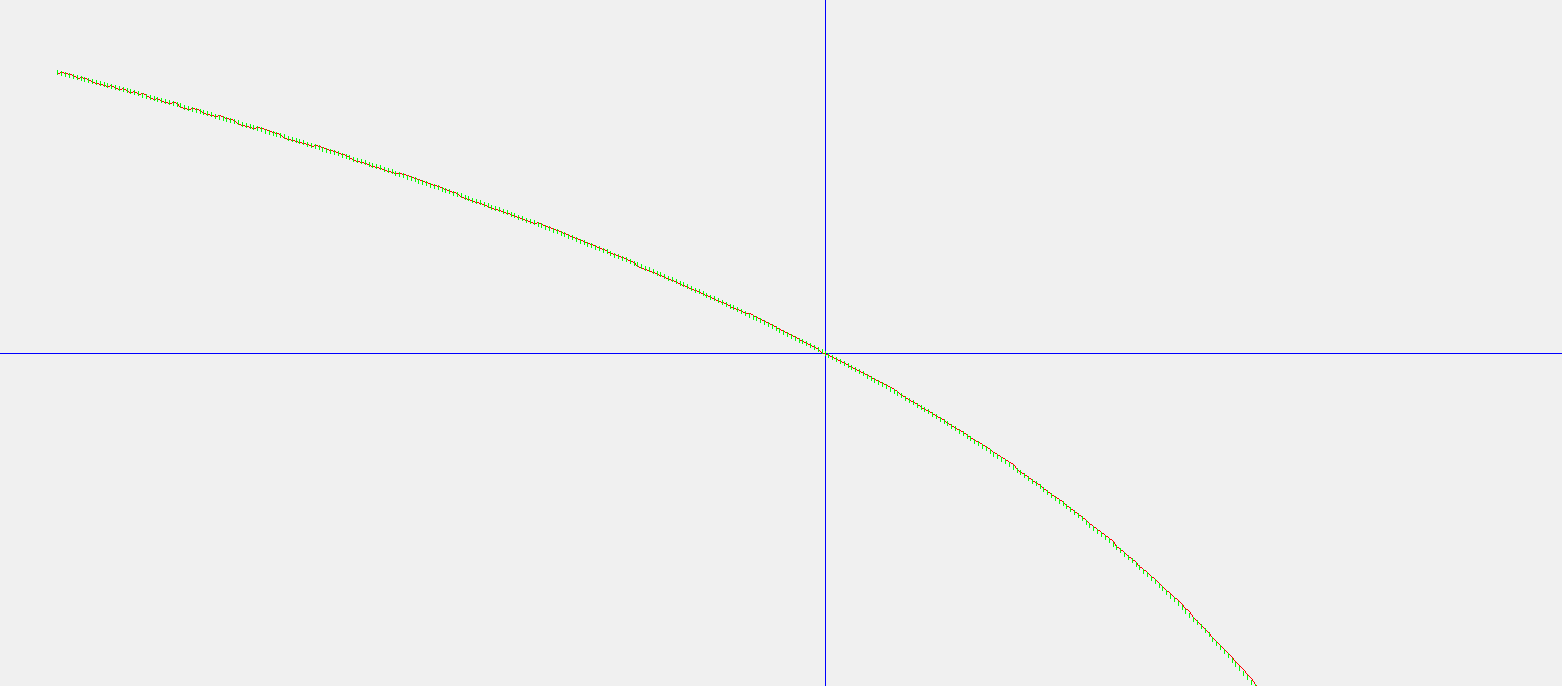
Код класса MainApp:

namespace lab1;  
  
public class MainApp  
{  
 private const string *fileOfMyFoo* = "C:\\Users\\Gutu\\RiderProjects\\Grafica\_lab1\\lab1\\FileOfMyFoo.txt";  
 private const string *fileOfAbsolute* = "C:\\Users\\Gutu\\RiderProjects\\Grafica\_lab1\\lab1\\FileToCheckAbsolute.txt";  
  
 [STAThread]  
 public static void *Main*()  
 {  
 MyService.*ClearFile*(*fileOfMyFoo*);  
 MyService.*ClearFile*(*fileOfAbsolute*);  
 MainDialog mainDialog = new MainDialog();  
 mainDialog.ShowDialog();  
 }  
}

Код класса MyService ответственного за запись и чтение значений в файл:

using Exception = System.Exception;  
  
namespace lab1;  
  
public class MyService  
{  
 public static void *WriteLineToFile*(string line, string filePath)  
 {  
 using (StreamWriter writer = new StreamWriter(filePath, true))  
 {  
 writer.WriteLine(line);  
 }  
  
   
 }  
  
 public static void ClearFile(string filePath)  
 {  
 using (FileStream fileStream = File.*Open*(filePath, FileMode.*Create*))  
 {  
 fileStream.SetLength(0);  
 }  
 }  
  
 public static string[] *ReadLinesFromFile*(string filePath)  
 {  
 try  
 {  
 string[] lines = File.*ReadAllLines*(filePath);  
 return lines;  
 }  
 catch (IOException e)  
 {  
 Console.*WriteLine*(e);  
 throw;  
 }  
 }  
}

## Результат отработки программы



Вывод окна MainDialog. Как видно график заданный при помощи разложения и встроенных библиотек совпадают.

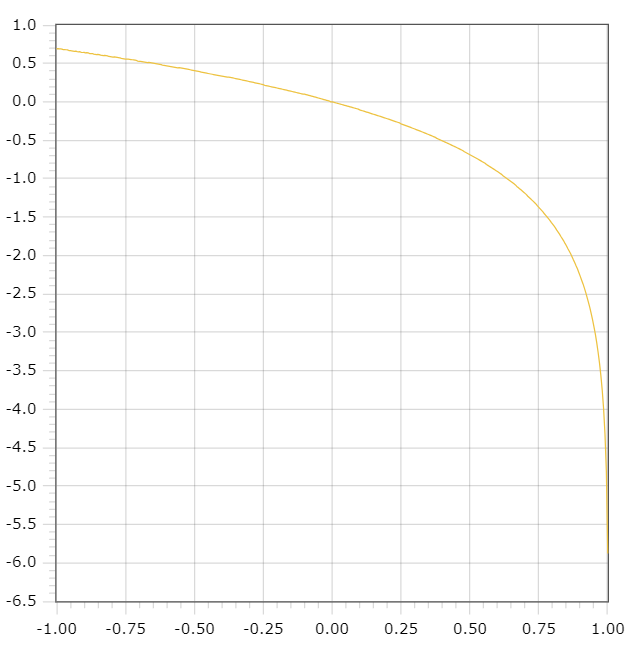
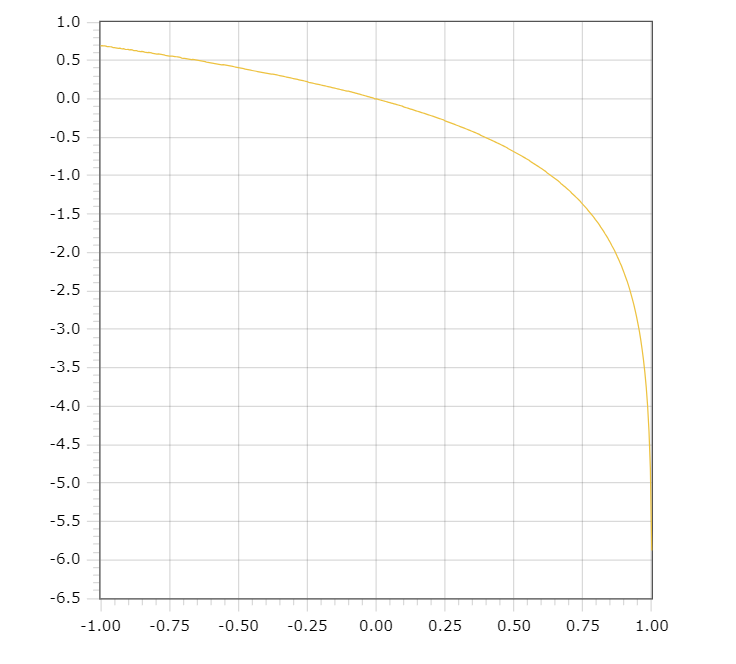
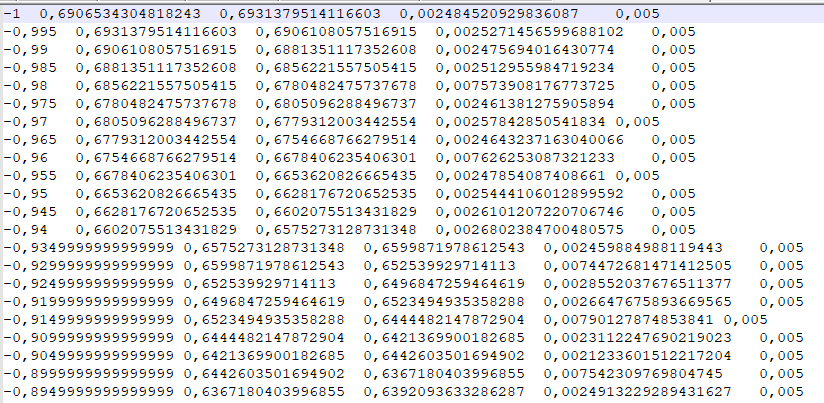
 

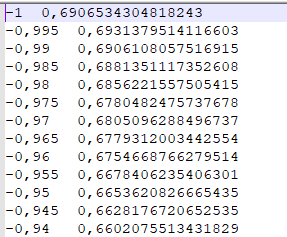
График построенный аналитически и график, заданный с точек, записанных в файл, совпадают.

Файл с записями значений функции с помощью встроенной библиотек, путем разложения и их разность:



Как видно по записям погрешность в вычислениях при помощи разложения чаще всего не превышает заданную погрешность.

Файл со значениями функции, вычисленными путем разложения:



# ВЫВОД

В ходе выполнения работы были изучены базовые составляющие графики на языке программирования C#. Были изучены средства для построения оконных приложений с графиками функции. Так же затронуты такие пункты, как обработка событий, чтение и запись данных в файл.