**Міністерство освіти і науки України**

**Національний університет «Львівська політехніка»**

**Інститут прикладної математики та фундаментальних наук**

**Кафедра прикладної математики**

**Звіт**

до лабораторної роботи №4

Варіант - 7

Виконала:

студентка групи

ПМ-32

Яворська Олена

Прийняв:

Строчик М. М.

Львів 2020

**Загальні положення**

Використовуючи об’єкти типу *DataGridView*та *PictureBox*, побудувати:

A) графік явно заданої функції;

B) графік функції, заданої параметрично.

На формі необхідно розташувати компоненти *DataGridView*та форму для малювання, поля для введення меж зміни аргументу, поле для введення кількості точок табулювання, поля для відображення найбільшого та найменшого значень функції на заданому інтервалі. У таблицю *DataGridView*необхідно вивести значення аргументу (аргументів) і відповідні табульовані значення функції. На графіках необхідно відобразити осі координат з позначками значень та відповідними підписами. Елементи керування, розташовані на формі, повинні коректно відображуватися при зміні розмірів форми. Передбачити можливість як зчитування даних з файлу для побудови графіків, так і збереження табульованих значень та графіків у відповідні файли. Додатковими опціями можуть бути сітка з можливістю її відображення/приховування; побудова графіка для діапазону значень, виділених користувачем у таблиці *DataGridView*; масштабування графіка.

### Завдання:

### Графік явно заданої функції:

### y=exp(−2x)sin^2x

### Для параметрично заданої функції побудувати графіки функцій x(t),y(t),y(x)x(t),y(t),y(x) та x(y):

### x(t)=t^1/3,

### y(t)=(1−t)^1/3

### Хід роботи:

### Як і у лабораторній 3, використала TabPages, тільки цього разу замість чотирьох графіків для параметричної функції побудувала один (менше повторень в прописуванні таблиці даних, тільки для іншої вкладки) .

### Познайомилася з формою для малювання, розібралася із збереженням Bitmap( сам графічний об’єкт з пікчербоксу зберегти теоретично можливо, знаходила , на практиці не працює) та його відображенням (нюанс, через який не було зрозуміло, чому рисунок зберігається правильно, але в пікчербоксі нічого не промальовується).

### Передбачила як зчитування даних з файлу для побудови графіків, так і збереження табульованих значень та графіків у відповідні файли.

### Збереження графіків:

private void button2\_Click(object sender, EventArgs e)

{

bmp.Save(@"plot1.PNG", System.Drawing.Imaging.ImageFormat.Png);

bmp2.Save(@"plot2.PNG", System.Drawing.Imaging.ImageFormat.Png);

//plot1.Image.Save(@"plot1.PNG", System.Drawing.Imaging.ImageFormat.Png);

//plot2.Image.Save(@"plot2.PNG", System.Drawing.Imaging.ImageFormat.Png);

}

### Збереження таблиць значень( як з попередньої лр, тільки дві):

### private void button1\_Click\_1(object sender, EventArgs e)

//записуємо дані з кожного "datagridview в окремий файл

{

string docPath = Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);

StreamWriter sw1 = new StreamWriter(Path.Combine(docPath, "datagridview1.txt"));

StreamWriter sw2 = new StreamWriter(Path.Combine(docPath, "datagridview2.txt"));

string point = ";";//розділювач

for (int i = 1; i < dataGridView1.RowCount; i++)

{

for (int j = 0; j < dataGridView1.ColumnCount; j++)

{

sw1.Write(Convert.ToString(dataGridView1[j, i].Value + point));

}

}

for (int i = 1; i < dataGridView2.RowCount; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

sw2.Write(Convert.ToString(dataGridView2[j, i].Value + point));

}

}

//закриваємо файли,куди записували

sw1.Close();

sw2.Close();

# }

### Зчитування даних про відрізок та к-ість його розбиттів з файла:

//можливість отримати дані про відрізок та к-ість проміжків розбиття з файлу

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

FileInfo f = new FileInfo(@"C:\Users\Olena\source\repos\lab3\_yavO\lab3\_yavO\data.txt");//розташування моєї лр

using (FileStream fs = f.Open(FileMode.Open, FileAccess.Read, FileShare.ReadWrite))

{

using (StreamReader sr = new StreamReader(fs, Encoding.UTF8))//зчитує файл з даними

{

from.Text = sr.ReadLine();//перший рядок з файлу

to.Text = sr.ReadLine();//другий,де я записала кінець відрізку

number.Text = sr.ReadLine();//відповідно,третій рядок мого файлу

}

}

### }

Всі графіки/текстові файли зі значеннями іксів і функцій в них в архіві з програмою.

### 3)На графіках необхідно відобразила осі координат з позначками значень та відповідними підписами.

private void pidpys(double []x, double[] y,Graphics graphics)

{

// Create point for upper-left corner of drawing.

float point\_y = 300.0F;

float point\_x = 0.0F;

for (float i = 0; i <= (x.Length) ; i++)

{

// Draw string to screen.

graphics.DrawString((i).ToString(), drawFont, drawBrush, i \* 15.0F, point\_y, drawFormat);

//int wX = plot1.Width;

//int hX = plot1.Height;

}

for (float i = 0; i <= (y.Length); i++)

{

graphics.DrawString((i).ToString(), drawFont, drawBrush, point\_x, 300.0F-(i \* 15.0F), drawFormat);

}

String drawString\_x = "x";

String drawString\_y= "F";

graphics.DrawString(drawString\_x, drawFont, drawBrush, 250.0F,320.0F, drawFormat);

graphics.DrawString(drawString\_y, drawFont, drawBrush, 20.0F, 0.0F, drawFormat);

### }

4) Передбачила оновлення даних при побудові графіка на новому відрізку DataGridView. Дозволила ввід в текстбокси лише цифр. При обраному RadioButton ці цифри беруться з мого файлу. Встановила дефолтні значення для текстбоксів з даними про відрізок та к-ість точок табулювання.

### 5) Для графіка 2(де 4 функції, параметричн) додала легенду:

//ЛЕГЕНДА ДЛЯ ГРАФІКА 2 (БО ТАМ НЕЧІТКО)

String drawString\_1 = "x(t)";

String drawString\_2 = "y(t)";

String drawString\_3 = "x(y)";

String drawString\_4 = "y(x)";

gr2.DrawLine(myPen, new Point(200, 130), new Point(220, 130));

gr2.DrawLine(myPen1, new Point(200, 150), new Point(220, 150));

gr2.DrawLine(myPen2, new Point(200, 170), new Point(220, 170));

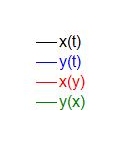
gr2.DrawLine(myPen3, new Point(200, 190), new Point(220, 190));

gr2.DrawString(drawString\_1, drawFont1, drawBrush, 220.0F, 120.0F, drawFormat);

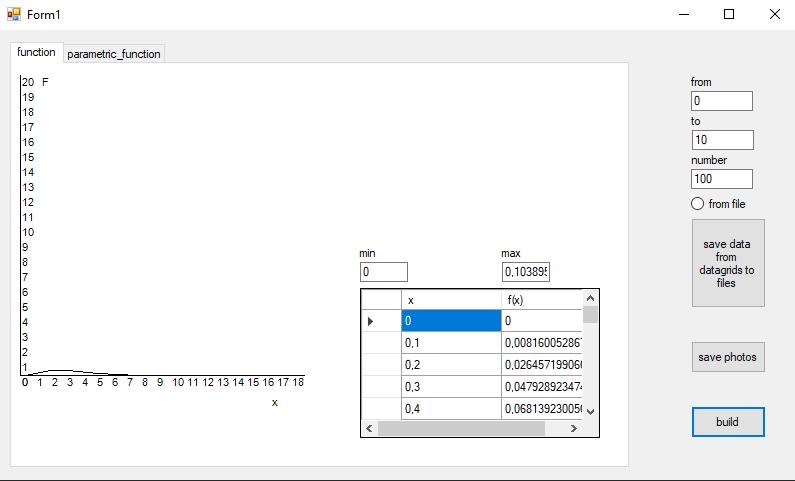
gr2.DrawString(drawString\_2, drawFont1, drawBrush1, 220.0F, 140.0F, drawFormat);

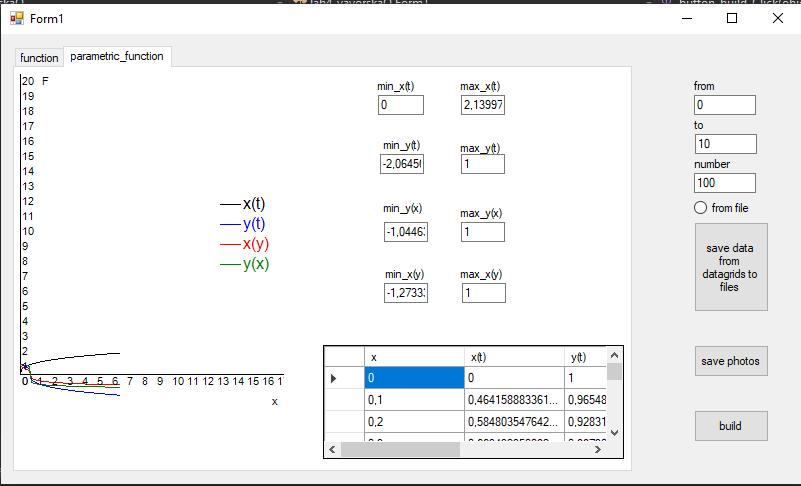
gr2.DrawString(drawString\_3, drawFont1, drawBrush2, 220.0F, 160.0F, drawFormat);

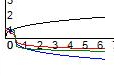
### gr2.DrawString(drawString\_4, drawFont1, drawBrush3, 220.0F, 180.0F, drawFormat);

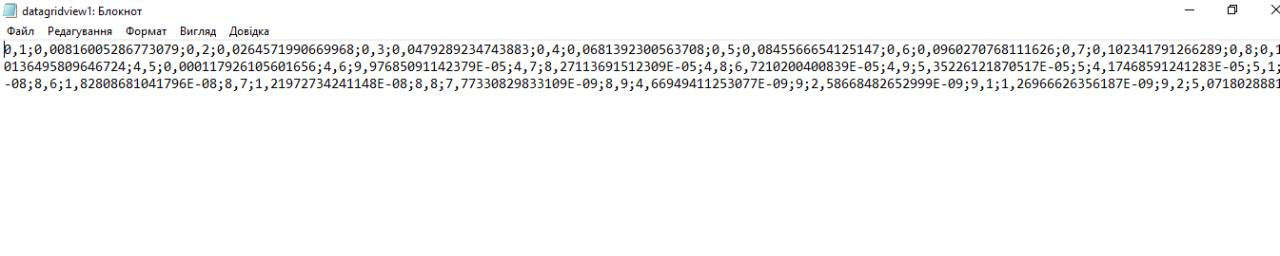


**Результат виконання програми:**





****



Тут(вище) показано, як збереглися дані з DataGridView.

**Висновок:** на цій лабораторній роботі вперше працювала з графіками та зчитуванням з/записом в файл збереженням зображень в елементі форми PictureBox. Графіки з результати виконання програми вище можуть виглядати трохи неакуратно, і , принципі, графік параметрично заданої функції можна було б масштабувати, але , якщо порівнювати з графіками з лр 3, це ті ж самі функції/графіки, просто відображені трохи по інакшому… Наприклад, якщо в об’єкті Chart можна було зробити осі по х,наприклад, від 1 до 10, і по у від 0 до 2, то тут осі мають однакові розмітки, хоч я і будувала їх на основі масивів точок зі значеннями х та функцій в точках. Тобто графіки можуть трохи по іншому виглядати через співвідношення осей, але це одні й ті ж графіки, що й в лр 3.

Могли б виникнути проблеми з явним заданням деяких точок, але , оскільки моя форма має статичний розмір і не може його змінювати, їх не виникає.

**Додаток:**

Program.cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace lab4\_yavorskaO

{

static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

Form1.cs:

using System;

using System.Collections;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

//using System.Windows.Forms.DataVisualization.Charting;

using System.IO;//бібліотека для читання-запису в файл

namespace lab4\_yavorskaO

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

//дефолтні значення при загрузці,їх можна змінити ,але лише на цифри,далі буде обмеження вводу

//у відповідні текстбокси

number.Text = "100";

from.Text = "0";

to.Text = "10";

//gr = plot1.CreateGraphics();

//gr2 = plot2.CreateGraphics();

//створюємо об'єкти , де будемо малювати

bmp = new Bitmap(plot1.Width, plot1.Height);

gr = Graphics.FromImage(bmp);

bmp2 = new Bitmap(plot1.Width, plot1.Height);

gr2 = Graphics.FromImage(bmp2);

}

Bitmap bmp;

Bitmap bmp2;

Graphics gr;

Graphics gr2;

Font drawFont = new Font("Arial", 8);

Font drawFont1 = new Font("Arial", 12);

SolidBrush drawBrush = new SolidBrush(Color.Black);

SolidBrush drawBrush1 = new SolidBrush(Color.Blue);

SolidBrush drawBrush2 = new SolidBrush(Color.Red);

SolidBrush drawBrush3 = new SolidBrush(Color.Green);

StringFormat drawFormat = new StringFormat();

Pen myPen = new Pen(Color.Black, 1.5f);

Pen myPen1 = new Pen(Color.Blue, 1.5f);

Pen myPen2 = new Pen(Color.Red, 1.5f);

Pen myPen3 = new Pen(Color.Green, 1.5f);

private void label4\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{//кнопка save data from datagrids to files

//записуємо дані з кожного "datagridview в окремий файл

{

string docPath = Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);

StreamWriter sw1 = new StreamWriter(Path.Combine(docPath, "datagridview1.txt"));

StreamWriter sw2 = new StreamWriter(Path.Combine(docPath, "datagridview2.txt"));

string point = ";";//розділювач

for (int i = 1; i < dataGridView1.RowCount; i++)

{

for (int j = 0; j < dataGridView1.ColumnCount; j++)

{

sw1.Write(Convert.ToString(dataGridView1[j, i].Value + point));

}

}

for (int i = 1; i < dataGridView2.RowCount; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

sw2.Write(Convert.ToString(dataGridView2[j, i].Value + point));

}

}

}

}

private void button\_build\_Click(object sender, EventArgs e)

{

//dataGridView1.Refresh();

//очищаємо dataGridView,щоб при повторному натисканні на button\_build будувалася заново таблиця значень

//якщо робити лише dataGridView1.Refresh(),то нові значення додаватимуться після старих

//якщо нічого не робити dataGridView не зміниться при зміні меж і графіків

dataGridView1.Rows.Clear();

dataGridView2.Rows.Clear();

//очищаємо пікчербокси

gr.Clear(Color.White);

gr2.Clear(Color.White);

double Xmin = double.Parse(from.Text);

double Xmax = double.Parse(to.Text);

//якщо введуть початок відрізка,більший за кінець

if (Xmin > Xmax)

{

double temp;

temp = Xmin;

Xmin = Xmax;

Xmax = temp;

}

int count = int.Parse(number.Text);//к-ість відрізків розбиття

//int count = 0;

//if (int.TryParse(number.Text, out count)) ;

double Step;//крок,з яким рахуємо значення ф-ії в точках

Step = (Math.Abs(Xmax - Xmin)) / count;//модуль,щоб не вибивало помилки,коли введуть поч відр ,більший за кінець

// Массив значень X – спільний для обох графіківв

double[] x = new double[count];

// Масиви Y – по одному для кожного графіка

double[] y1 = new double[count];

double[] y2 = new double[count];

double[] y3 = new double[count];

double[] y4 = new double[count];

double[] y5 = new double[count];

// рахуєм точки для графіків функцій в циклі

for (int i = 0; i < count; i++)

{// вичисляємо Х на кожному кроці

x[i] = Xmin + Step \* i;

// вичисляємо значення функцій в точках X

//x(t)=t^1/3---->sign(t)\*(abs(t))^1/3

y1[i] = Math.Exp(-2 \* x[i]) \* Math.Pow(Math.Sin(x[i]), 2);//графік явно заданої функції

y2[i] = Math.Sign(x[i]) \* Math.Pow(Math.Abs(x[i]), 1.0 / 3.0);//x(t) для параметрично заданої функції

y3[i] = Math.Sign(1 - x[i]) \* Math.Pow(Math.Abs(1 - x[i]), 1.0 / 3.0);//y(t)

y5[i] = Math.Sign(y3[i]) \* Math.Pow(Math.Abs(y3[i]), 1.0 / 3.0);//x(y) для параметрично заданої функції

y4[i] = Math.Sign(1 - y2[i]) \* Math.Pow(Math.Abs(1 - y2[i]), 1.0 / 3.0);//y(x)

//додаємо виведення іксів і значень функцій в них з кожного графіка у відповфдну dataGridView

dataGridView1.Rows.Add(x[i], y1[i]);

dataGridView2.Rows.Add(x[i], y2[i], y3[i], y4[i], y5[i]);

}

int scale = Convert.ToInt32(250 \* 2 / (Xmax - Xmin));

//int scale1 = Convert.ToInt32(50 \* 2 / (Xmax - Xmin));

int scale1 = Convert.ToInt32(50 \* 2 / (Xmax - Xmin));

// МАЛЮВАННЯ ОСЕЙ

if (Xmin < 0)

{

gr.DrawLine(myPen, new Point(Convert.ToInt32(-Xmin \* scale), 100), new Point(Convert.ToInt32(-Xmin \* scale), 300));//вісь у

gr.DrawLine(myPen, new Point(0, 300), new Point(500, 300));//вісь х

gr2.DrawLine(myPen, new Point(Convert.ToInt32(-Xmin \* scale), 100), new Point(Convert.ToInt32(-Xmin \* scale), 300));//вісь у

gr2.DrawLine(myPen, new Point(0, 300), new Point(500, 300));//вісь х

}

else

{

gr.DrawLine(myPen, new Point(0, 0), new Point(0, 300));//вісь у

gr.DrawLine(myPen, new Point(0, 300), new Point(500, 300));//вісь х

gr2.DrawLine(myPen, new Point(0, 0), new Point(0, 300));//вісь у

gr2.DrawLine(myPen, new Point(0, 300), new Point(500, 300));//вісь х

}

//ВЛАСНЕ МАЛЮВАННЯ ГРАФІКІВ

for (int i = 0; i < x.Length - 1; i++)

{

gr.DrawLine(myPen, new Point(Convert.ToInt32((x[i] - Xmin) \* scale), (-1) \* Convert.ToInt32(y1[i] \* scale) + 300), new Point(Convert.ToInt32((x[i + 1] - Xmin) \* scale), (-1) \* Convert.ToInt32(y1[i + 1] \* scale) + 300));

gr2.DrawLine(myPen, new Point(Convert.ToInt32((x[i] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y2[i] \* scale1) + 300), new Point(Convert.ToInt32((x[i + 1] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y2[i + 1] \* scale1) + 300));

gr2.DrawLine(myPen1, new Point(Convert.ToInt32((x[i] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y3[i] \* scale1) + 300), new Point(Convert.ToInt32((x[i + 1] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y3[i + 1] \* scale1) + 300));

gr2.DrawLine(myPen2, new Point(Convert.ToInt32((x[i] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y4[i] \* scale1) + 300), new Point(Convert.ToInt32((x[i + 1] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y4[i + 1] \* scale1) + 300));

gr2.DrawLine(myPen3, new Point(Convert.ToInt32((x[i] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y5[i] \* scale1) + 300), new Point(Convert.ToInt32((x[i + 1] - Xmin) \* scale1), (-1) \* Convert.ToInt32(y5[i + 1] \* scale1) + 300));

}

//ЛЕГЕНДА ДЛЯ ГРАФІКА 2 (БО ТАМ НЕЧІТКО)

String drawString\_1 = "x(t)";

String drawString\_2 = "y(t)";

String drawString\_3 = "x(y)";

String drawString\_4 = "y(x)";

gr2.DrawLine(myPen, new Point(200, 130), new Point(220, 130));

gr2.DrawLine(myPen1, new Point(200, 150), new Point(220, 150));

gr2.DrawLine(myPen2, new Point(200, 170), new Point(220, 170));

gr2.DrawLine(myPen3, new Point(200, 190), new Point(220, 190));

gr2.DrawString(drawString\_1, drawFont1, drawBrush, 220.0F, 120.0F, drawFormat);

gr2.DrawString(drawString\_2, drawFont1, drawBrush1, 220.0F, 140.0F, drawFormat);

gr2.DrawString(drawString\_3, drawFont1, drawBrush2, 220.0F, 160.0F, drawFormat);

gr2.DrawString(drawString\_4, drawFont1, drawBrush3, 220.0F, 180.0F, drawFormat);

//gr2.DrawString(drawString\_2, drawFont, drawBrush, 20.0F, 0.0F, drawFormat);

//підписуємо осі

pidpys(x, y1, gr);

pidpys(x, y1, gr2);

//заповняємо min i max для графіка 1

double y1\_min = func\_min(y1, count);

min.Text = y1\_min.ToString();

double y1\_max = func\_max(y1, count);

max.Text = y1\_max.ToString();

//заповняємо min i max для графіка 2

double y2\_min = func\_min(y2, count);

min\_2.Text = y2\_min.ToString();

double y2\_max = func\_max(y2, count);

max\_2.Text = y2\_max.ToString();

//заповняємо min i max для графіка 3

double y3\_min = func\_min(y3, count);

min\_3.Text = y3\_min.ToString();

double y3\_max = func\_max(y3, count);

max\_3.Text = y3\_max.ToString();

//заповняємо min i max для графіка 4

double y4\_min = func\_min(y4, count);

min\_4.Text = y4\_min.ToString();

double y4\_max = func\_max(y4, count);

max\_4.Text = y4\_max.ToString();

//заповняємо min i max для графіка 5

double y5\_min = func\_min(y5, count);

min\_5.Text = y5\_min.ToString();

double y5\_max = func\_max(y5, count);

max\_5.Text = y5\_max.ToString();

//вивільняємо

//gr.Dispose();

//gr2.Dispose();

//відображає наші об'єкти бітмап на формі(якщо цього не зробити, в файл

//зберігатимуться правильні рисунки, натомість на формі буде пустий пікчербокс

plot1.Image = bmp;

plot2.Image = bmp2;

}

private void pidpys(double []x, double[] y,Graphics graphics)

{

// Create point for upper-left corner of drawing.

float point\_y = 300.0F;

float point\_x = 0.0F;

for (float i = 0; i <= (x.Length) ; i++)

{

// Draw string to screen.

graphics.DrawString((i).ToString(), drawFont, drawBrush, i \* 15.0F, point\_y, drawFormat);

//int wX = plot1.Width;

//int hX = plot1.Height;

}

for (float i = 0; i <= (y.Length); i++)

{

graphics.DrawString((i).ToString(), drawFont, drawBrush, point\_x, 300.0F-(i \* 15.0F), drawFormat);

}

String drawString\_x = "x";

String drawString\_y= "F";

graphics.DrawString(drawString\_x, drawFont, drawBrush, 250.0F,320.0F, drawFormat);

graphics.DrawString(drawString\_y, drawFont, drawBrush, 20.0F, 0.0F, drawFormat);

}

private void tabPage1\_Click(object sender, EventArgs e)

{

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

FileInfo f = new FileInfo(@"C:\Users\Olena\source\repos\lab3\_yavO\lab3\_yavO\data.txt");//розташування моєї лр

using (FileStream fs = f.Open(FileMode.Open, FileAccess.Read, FileShare.ReadWrite))

{

using (StreamReader sr = new StreamReader(fs, Encoding.UTF8))//зчитує файл з даними

{

from.Text = sr.ReadLine();//перший рядок з файлу

to.Text = sr.ReadLine();//другий,де я записала кінець відрізку

number.Text = sr.ReadLine();//відповідно,третій рядок мого файлу

}

}

}

// всі KeyPress призн,щоб не дати ввести в текстбокси щось,крім цифр

private void from\_KeyPress(object sender, KeyPressEventArgs e)

{

char number = e.KeyChar;

if (!Char.IsDigit(number) && number != 8 && number != 44 && e.KeyChar != '-') // цифри, клавиша BackSpace и кома

{

e.Handled = true;

}

}

private void to\_KeyPress(object sender, KeyPressEventArgs e)

{

char number = e.KeyChar;

if (!Char.IsDigit(number) && number != 8 && number != 44 && e.KeyChar != '-') // цифри, клавиша BackSpace и кома

{

e.Handled = true;

}

}

private void number\_KeyPress(object sender, KeyPressEventArgs e)

{

char number = e.KeyChar;

if (!Char.IsDigit(number) && number != 8 && number != 44 && e.KeyChar != '-') // цифри, клавиша BackSpace и кома

{

e.Handled = true;

}

}

private void plot2\_Click(object sender, EventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e)

{

//radioButton1.Checked = false;

//початково нема читання з ф

}

private void plot2\_Click\_1(object sender, EventArgs e)

{

}

private void plot1\_Paint(object sender, PaintEventArgs e)

{

}

//функція для збереження графіків у файл

private void button2\_Click(object sender, EventArgs e)

{

//plot1.Image.Save(@"F:\chart1.png", System.Drawing.Imaging.ImageFormat.Jpeg);

bmp.Save(@"plot1.PNG", System.Drawing.Imaging.ImageFormat.Png);

bmp2.Save(@"plot2.PNG", System.Drawing.Imaging.ImageFormat.Png);

//plot1.Image.Save(@"plot1.PNG", System.Drawing.Imaging.ImageFormat.Png);

//plot2.Image.Save(@"plot2.PNG", System.Drawing.Imaging.ImageFormat.Png);

}

//функція для визнач min f(x)

private double func\_min(double[] y, double count)

{

double y\_min = y[0];

for (int i = 0; i < count - 1; i++)

{

if (y[i] < y\_min)

{

y\_min = y[i];

}

}

return y\_min;

}

//функція для визнач максимуму f(x)

private double func\_max(double[] y, double count)

{

double y\_max = y[0];

for (int i = 0; i < count - 1; i++)

{

if (y[i] > y\_max)

{

y\_max = y[i];

}

}

return y\_max;

}

}

}