**ЛАБОРАТОРНА РОБОТА № 8**

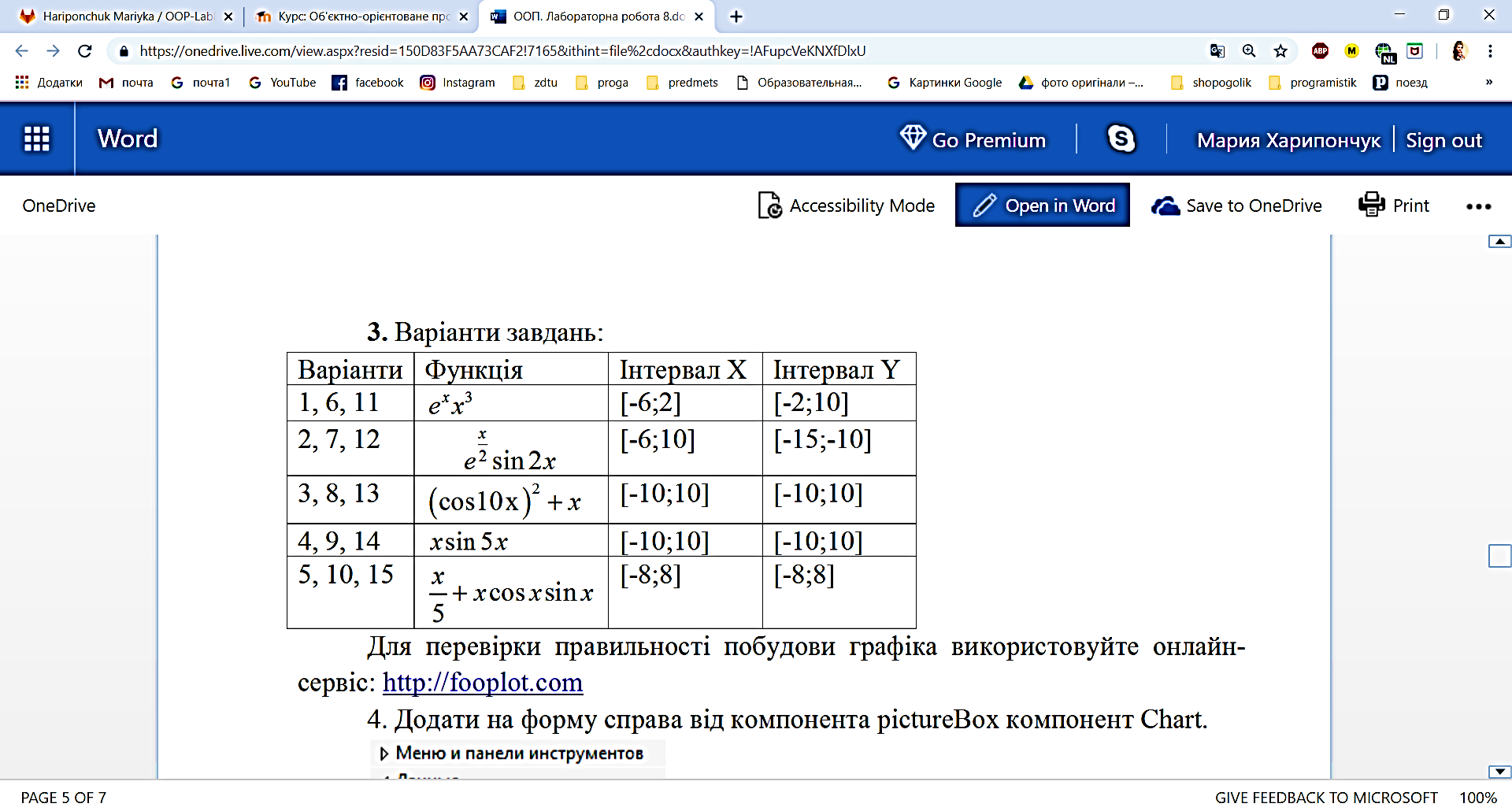
**Використання графічних можливостей C#.**

***Мета:*** засвоїти принципи малювання графічних примітивів за допомогою .NET Framework.

**Завдання на лабораторну роботу:**

**1**. Створити рішення з назвою «oop-lab8».

Створити проект Windows Forms з назвою «GraphicPlotter».

**2**. Завдання передбачає написання програмного коду для побудови графіка заданої у варіанті функції. При зміні розмірів вікна графік повинен масштабуватися.

Лістинг програми:

using System;

using System.Drawing;

using System.Windows.Forms;

namespace GraphicPlotter

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Paint(object sender, PaintEventArgs e)

{

//one();

}

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

{

//one();

}

void one()

{

Graphics g = this.CreateGraphics(); //задаем область рисования

g.Clear(Color.White); //очищаем область (заливаем белым цветом)

Pen axis = new Pen(Color.Blue, 2); //ручка для рисования осей

Pen graph = new Pen(Color.Green, 3); //ручка для рисования графика

Font drawFont = new Font("Century Gothic", 13); //шрифт которым будем подписывать оси

Font signatureFont = new Font("Century Gothic", 7); //шрифт которым будем подписывать деления оси

SolidBrush drawBrush = new SolidBrush(Color.Blue); //цвет этого шрифта

StringFormat drawFormat = new StringFormat(); //формат букв (для подписывания)

drawFormat.FormatFlags = StringFormatFlags.DirectionRightToLeft; //напрявление текста оси

int sizeWidth = ActiveForm?.Width ?? 0; //ширина окна программы

int sizeHeight = ActiveForm?.Height ?? 0; //висота окна программы

//x, y - координаты цетра (точки 0)

Point center = new Point(((int)(sizeWidth / 2) - 8), (int)((sizeHeight / 2) - 19));

g.DrawLine(axis, 10, center.Y, center.X, center.Y); //ось X-

g.DrawLine(axis, center.X, center.Y, 2 \* center.X - 10, center.Y); //ось X+

g.DrawLine(axis, center.X, 10, center.X, center.Y); //ось Y+

g.DrawLine(axis, center.X, center.Y, center.X, 2 \* center.Y - 10); //ось Y-

g.DrawString("X", drawFont, drawBrush, new PointF(2 \* center.X - 5, center.Y + 10), drawFormat); //подписали x

g.DrawString("Y", drawFont, drawBrush, new PointF(center.X + 30, 5), drawFormat); //подписали y

g.DrawString("0", drawFont, drawBrush, new PointF(center.X, center.Y), drawFormat); //подписали 0

//стрелки:

g.DrawLine(axis, center.X, 10, center.X + 5, 20); //y+

g.DrawLine(axis, center.X, 10, center.X - 5, 20); //y-

g.DrawLine(axis, 2 \* center.X - 10, center.Y, 2 \* center.X - 20, center.Y + 5); //x+

g.DrawLine(axis, 2 \* center.X - 10, center.Y, 2 \* center.X - 20, center.Y - 5); //x-

int stepForAxes = 25; //расстояние между делениями на осях

int lenghtShtrih = 3; //половина длины штришка деления

int maxValueForAxesX = 9; //максимальное значение по оси X

int maxValueForAxesY = 16; //максимальное значение по оси Y

float oneDelenieX = (float)maxValueForAxesX / ((float)center.X / (float)stepForAxes); //то, чем подписывать деления X

float oneDelenieY = (float)maxValueForAxesY / ((float)center.Y / (float)stepForAxes); //то, чем подписывать деления Y

for (int i = center.X, j = center.X, k = 1; i < 2 \* center.X - 30; j -= stepForAxes, i += stepForAxes, k++)

{

g.DrawLine(axis, i, center.Y - lenghtShtrih, i, center.Y + lenghtShtrih); //рисуем штрихи по оси X вправо от центра

g.DrawLine(axis, j, center.Y - lenghtShtrih, j, center.Y + lenghtShtrih); // рисуем штрихи по оси X влево от центра

if (i < 2 \* center.X - 55)

{

g.DrawString((k \* oneDelenieX).ToString("0.0"), signatureFont, drawBrush, new PointF(i + stepForAxes + 9, center.Y + 6), drawFormat); //подписываем деления +

g.DrawString(((k \* oneDelenieX).ToString("0.0").ToString() + "-"), signatureFont, drawBrush, new PointF(j - stepForAxes + 9, center.Y + 6), drawFormat); //подписываем деления -

}

}

for (int i = center.Y, j = center.Y, k = 1; i < 2 \* center.Y - 30; j -= stepForAxes, i += stepForAxes, k++)

{

g.DrawLine(axis, center.X - lenghtShtrih, i, center.X + lenghtShtrih, i); //рисуем штрихи по оси Y вправо от центра

g.DrawLine(axis, center.X - lenghtShtrih, j, center.X + lenghtShtrih, j); // рисуем штрихи по оси Y влево от центра

if (i < 2 \* center.Y - 55)

{

g.DrawString(((k \* oneDelenieY).ToString("0.0").ToString() + "-"), signatureFont, drawBrush, new PointF(center.X + stepForAxes, i + stepForAxes - 6), drawFormat); //подписываем деления +

g.DrawString((k \* oneDelenieY).ToString("0.0"), signatureFont, drawBrush, new PointF(center.X + stepForAxes, j - stepForAxes - 6), drawFormat); //подписываем деления -

}

}

int numOfPoint = 100; //количество точек для расчета значения функции

float[] first = new float[numOfPoint]; //точки для расчета

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

{

first[i] = (float)maxValueForAxesX / (float)numOfPoint \* (i + 1) - (float)(maxValueForAxesX / 2); //интервал X от -8 до 8

first[i] = (float)maxValueForAxesY / (float)numOfPoint \* (i + 1) - (float)(maxValueForAxesY / 2); //интервал Y от -8 до 8

}

float[] second = new float[numOfPoint]; //значения в точках для расчета

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

second[i] = (float)((first[i] / 5) + first[i] \* Math.Cos(first[i]) \* Math.Sin(first[i])); //значение функции

Point[] pointOne = new Point[numOfPoint];

float tempX = 1 / oneDelenieX \* stepForAxes; //рассчитываем новые координаты

float tempY = 1 / oneDelenieY \* stepForAxes;

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

{

pointOne[i].X = center.X + (int)(first[i] \* tempX); //переход к новым координатам

pointOne[i].Y = center.Y - (int)(second[i] \* tempY);

}

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

{

chart1.Series[0].Points.AddXY(pointOne[i].X, pointOne[i].Y);

}

g.DrawCurve(graph, pointOne); //рисуем сплайном

}

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

{

one();

}

}

}

Результат виконання програми:

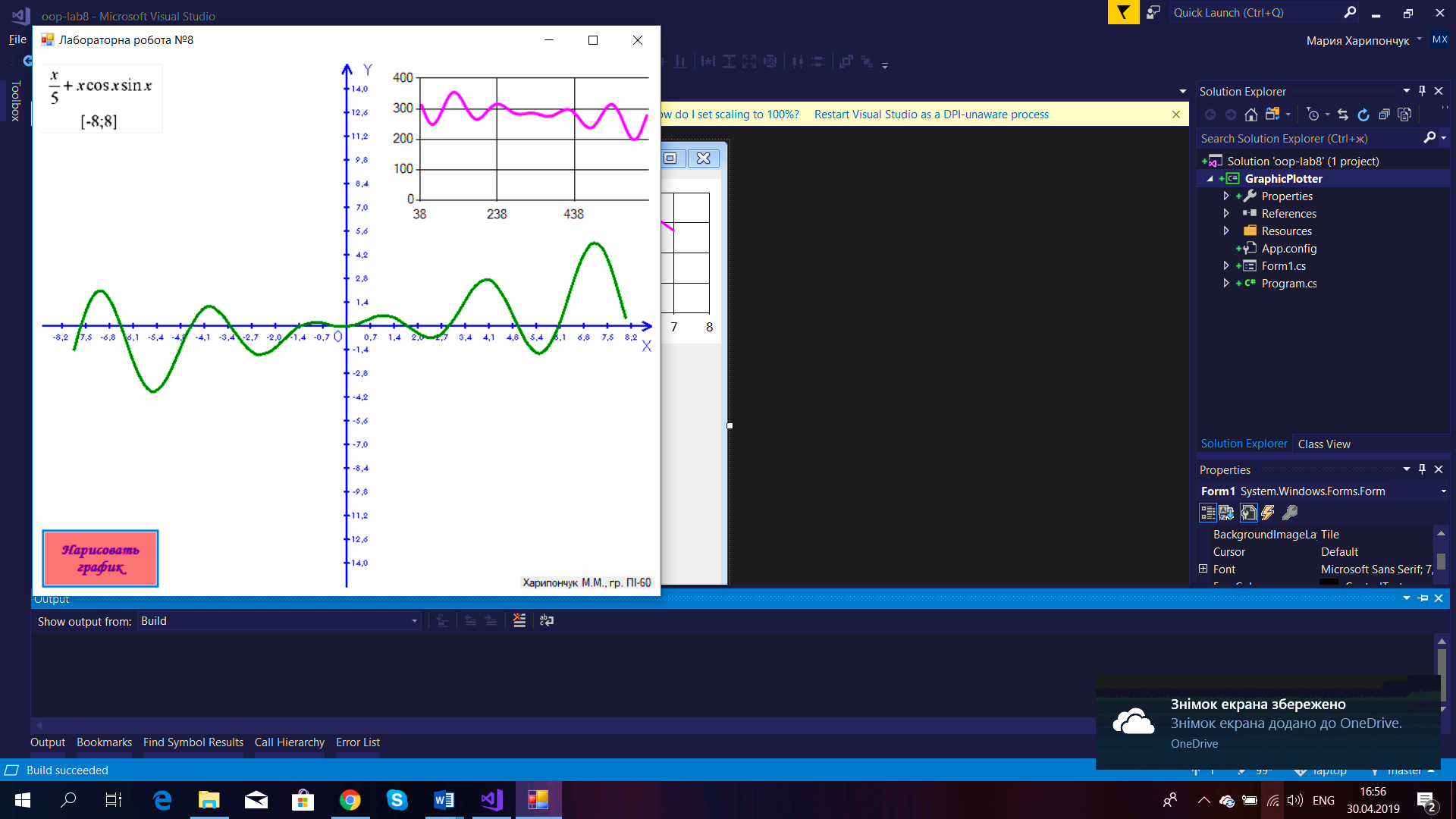


Рис. 1. Результат виконання роботи програми

***Висновок***: на даній лабораторній роботі було засвоєно принципи малювання графічних примітивів за до-помогою .NET Framework.