

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ ЭЛЕКТРОННО-ИНФОРМАЦИОННЫХ СИСТЕМ

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №7

Выполнила
Радиончик С.С.,
студентка группы ПО-5

Проверил
Крощенко А.А.,
ст. преп. Кафедры ИИТ,
«__» _____ 2021 г.

Брест, 2021

Цель работы: освоить возможности языка программирования C# в построении графических приложений.

Вариант 6.

Задание 1.

- Организовать ввод параметров для создания объектов (можно использовать файлы);
- Осуществить визуализацию графических примитивов, решить поставленную задачу

б) Задать движение окружности по апплету так, чтобы при касании границы окружность отражалась от нее.

Задание 2.

Реализовать построение заданного типа фрактала по варианту. Везде, где это необходимо, предусмотреть ввод параметров, влияющих на внешний вид фрактала.

б) Склоненное дерево Пифагора (обдуваемое ветром)

Выполнение:

Код программы.

1) *Form1.cs*

```
using System;
using System.Drawing;
using System.Windows.Forms;

namespace Lab7
{
    public partial class Form1 : Form
    {
        int diameter;
        int index;
        Graphics grph;

        int x = 10, y = 15;
        int dx = 10, dy = 10;

        public Form1()
        {
            InitializeComponent();
        }

        void DrawCircle(int x, int y)
        {
            int xc, yc;
            xc = x - diameter/2;
            yc = y - diameter /2;

            grph.FillEllipse(Brushes.Aqua, xc, yc, diameter/2, diameter/2);
            grph.DrawEllipse(Pens.Aqua, xc, yc, diameter/2, diameter/2);
        }

        public void Next()
        {
            if (x >= pictureBox1.Width - 2 * (diameter/2 + dx))
```

```

        dx = -dx;
    if (x < 0)
        dx = -dx;

    if (y >= pictureBox1.Height - SystemInformation.CaptionHeight - 2 *
(diameter/2 + dy))
        dy = -dy;
    if (y < 0)
        dy = -dy;

    x += dx;
    y += dy;
}

public void Move(Graphics g)
{
    g.Clear(pictureBox1.BackColor);
    Next();
    g.FillEllipse(Brushes.Aqua, x, y, diameter, diameter);
}

private void Form1_Load(object sender, EventArgs e)
{
    button1.Text = "Отобразить";
}

private void button1_Click(object sender, EventArgs e)
{
    index = 1;

    if(textBox1.Text != "")
        diameter = Convert.ToInt32(textBox1.Text);

    pictureBox1.Refresh();
}

private void pictureBox1_Paint(object sender, PaintEventArgs e)
{
    if (index == 1)
    {
        grph = pictureBox1.CreateGraphics();
        DrawCircle(x, y);
        timer1.Enabled = true;
    }
}

private void timer1_Tick(object sender, EventArgs e)
{
    Move(grph);
}
}
}

```

2) *Form1.cs*

```

using System;
using System.Drawing;
using System.Windows.Forms;

namespace Lab7_1
{
    public partial class Form1 : Form
    {
        Graphics grph;
        Bitmap map;
    }
}

```

```

Pen pen;

double angle = Math.PI / 2; //Угол поворота на 90 градусов
double ang1 = 0.7;
double ang2 = 0.9;

public Form1()
{
    InitializeComponent();
}

private void Form1_Load(object sender, EventArgs e)
{
    label1.Text = "Поворот левого ребенка:";
    label2.Text = "Поворот правого ребенка:";
    button1.Text = "Отобразить";
    pen = new Pen(Color.DarkSlateBlue);
}

private void button1_Click(object sender, EventArgs e)
{
    map = new Bitmap(pictureBox1.Width, pictureBox1.Height);
    grph = Graphics.FromImage(map);
    grph.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias; //сглаживание

    if (textBox1.Text != "")
        ang1 = Convert.ToDouble(textBox1.Text);

    if (textBox2.Text != "")
        ang2 = Convert.ToDouble(textBox2.Text);

    DrawTree(300, 450, 200, angle);

    pictureBox1.BackgroundImage = map;
}

//x и y - координаты родительской вершины
//a - параметр, который фиксирует количество итераций в рекурсии
//angle - угол поворота на каждой итерации
public int DrawTree(double x, double y, double a, double angle)
{
    if (a > 2)
    {
        a *= 0.7;

        double xnew = Math.Round(x + a * Math.Cos(angle)),
            ynew = Math.Round(y - a * Math.Sin(angle));

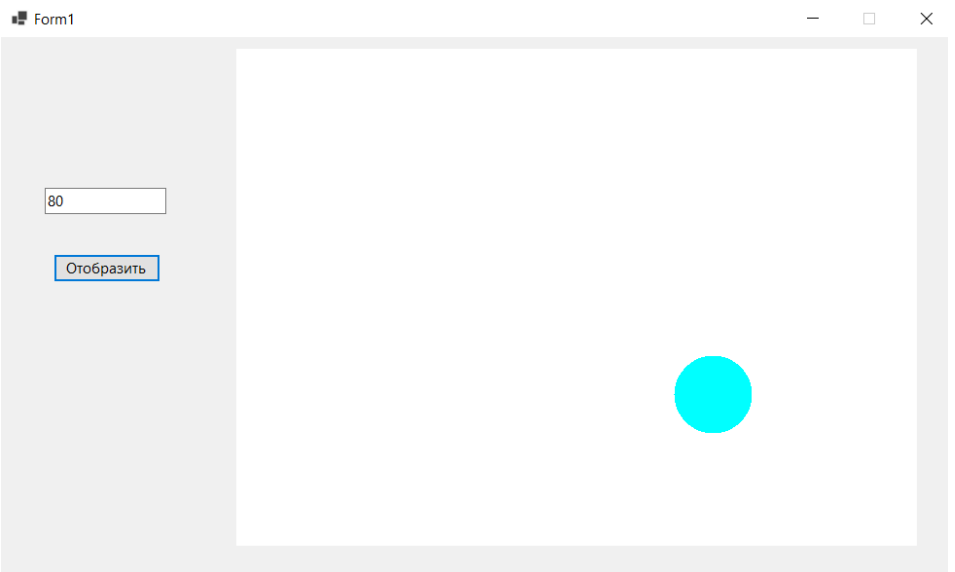
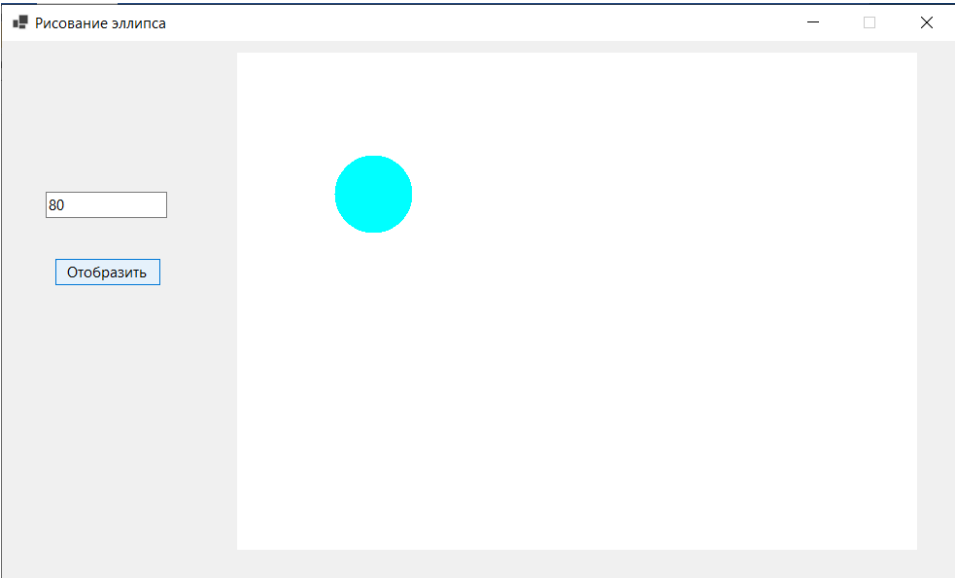
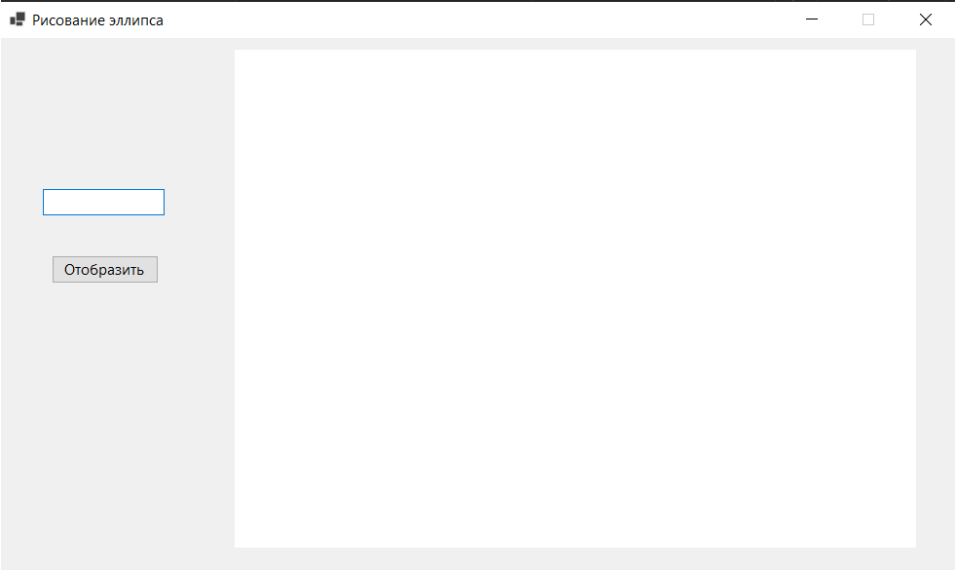
        grph.DrawLine(pen, (float)x, (float)y, (float)xnew, (float)ynew);

        x = xnew;
        y = ynew;

        //рекурсивная функция для левого и правого ребенка
        DrawTree(x, y, a, angle + ang1);
        DrawTree(x, y, a, angle - ang2);
    }
    return 0;
}
}
}

```

Результаты работы программы:
1)



2)

Form1

Поворот левого ребенка:

Поворот правого ребенка:

Отобразить

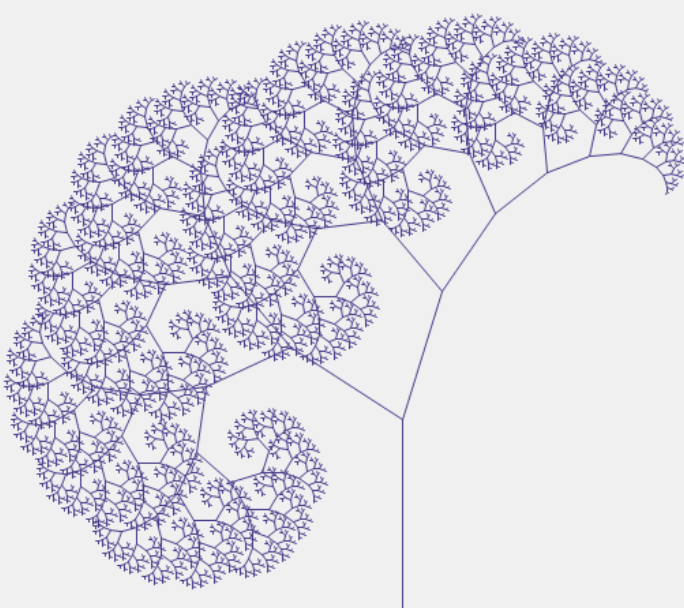


Form1

Поворот левого ребенка:

Поворот правого ребенка:

Отобразить



Вывод: освоила возможности языка программирования C# в построении графических приложений.