МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

 «БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра ИИТ

ЛАБОРАТОРНАЯ РАБОТА №7

Выполнил:

Студент ФЭИС

3-го курса, группы ПО-5

Белко В.А.

Проверил:

Крощенко А. А.

Брест 2021

**Цель работы:** приобрести навыки использования возможностей языка программирования C# в построении графических приложений.

**Задание 1**

Требования к выполнению

• Реализовать соответствующие классы, указанные в задании;

• Организовать ввод параметров для создания объектов (можно использовать файлы);

• Осуществить визуализацию графических примитивов, решить поставленную задачу

2) Определить класс Rectangle и класс Point. Объявить массив из n объектов класса Point. Написать

функцию, определяющую, какая из точек лежит снаружи, а какая – внутри прямоугольника**Код программы:**

using System;

using System.Collections.Generic;

using System.Drawing;

using System.Windows.Forms;

namespace spp\_lab7

{

namespace myGraph {

class Point

{

public Point(int x, int y)

{

X = x;

Y = y;

}

public readonly int X;

public readonly int Y;

}

class Rectangle

{

public Rectangle(Point point, int height, int width)

{

Point = point;

Height = height;

Width = width;

}

public readonly myGraph.Point Point;

public readonly int Height;

public readonly int Width;

public bool IsInside(Point elem)

{

bool param = false;

if (elem.X >= Point.X && elem.X <= Point.X + Height &&

elem.Y >= Point.Y && elem.Y <= Point.Y + Width)

{

param = true;

}

return param;

}

}

}

public partial class Task1 : Form

{

myGraph.Rectangle rectangle\_;

List<myGraph.Point> points\_ = new List<myGraph.Point>();

Pen pen\_ = new Pen(Color.Blue, 3);

public Task1()

{

InitializeComponent();

}

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

{

int X = 0;

int Y = 0;

int.TryParse(MyPointX.Text, out X);

int.TryParse(MyPointY.Text, out Y);

myGraph.Point Point = new myGraph.Point(X, Y);

if (rectangle\_?.IsInside(Point)??false)

{

points\_.Add(Point);

pen\_.Color = Color.Red;

using (Graphics Graph = PictB.CreateGraphics())

{

Graph.DrawRectangle(pen\_, Point.X, Point.Y, 1, 1);

}

Message.Text = "All right";

}

else

{

Message.Text = "Point is not in Rectangle!";

}

}

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

{

ClearAll();

int X = 0;

int Y = 0;

int Height = 0;

int Width = 0;

int.TryParse(RectPointX.Text, out X);

int.TryParse(RectPointY.Text, out Y);

int.TryParse(RectHeight.Text, out Height);

int.TryParse(RectWidth.Text, out Width);

myGraph.Point TLPoint = new myGraph.Point(X, Y);

pen\_.Color = Color.Blue;

rectangle\_ = new myGraph.Rectangle(TLPoint, Height, Width);

using (Graphics Graph = PictB.CreateGraphics())

{

Graph.DrawRectangle(pen\_, rectangle\_.Point.X, rectangle\_.Point.Y,

rectangle\_.Width, rectangle\_.Height);

}

}

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

{

ClearAll();

}

private void ClearAll()

{

using (Graphics Graph = PictB.CreateGraphics())

{

Graph.Clear(Color.LightGray);

points\_.Clear();

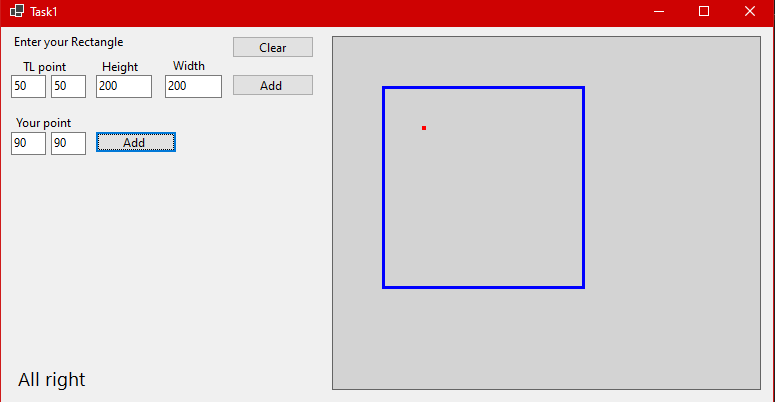
rectangle\_ = null;

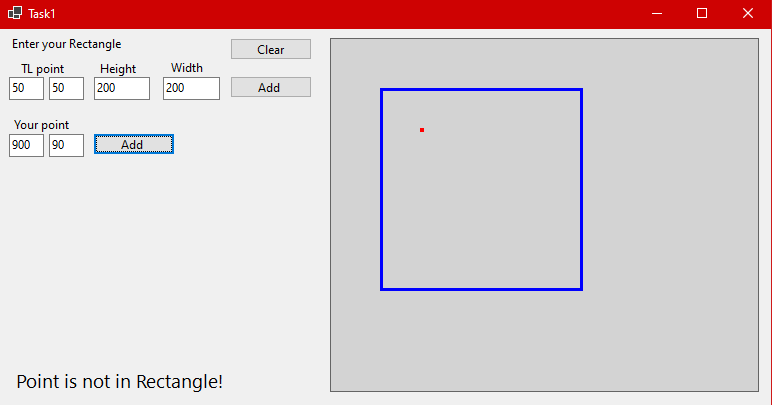
}

}

}

}**Вывод программы:**





**Задание 2**

Реализовать построение заданного типа фрактала по варианту

Везде, где это необходимо, предусмотреть ввод параметров, влияющих на внешний вид фрактала

2) Н-фрактал

**Код программы:**

using System;

using System.Drawing;

using System.Windows.Forms;

namespace task2

{

public partial class Task2 : Form

{

private int height\_;

private int width\_;

public Task2()

{

InitializeComponent();

height\_ = PictBoxTask2.Height;

width\_ = PictBoxTask2.Width;

}

private void DrawH(int x, int y, int size)

{

Pen myPen = new Pen(Color.Black, 1);

using (Graphics Graph = PictBoxTask2.CreateGraphics())

{

Graph.DrawLine(myPen, x - size, y - size, x - size, y + size);

Graph.DrawLine(myPen, x - size, y, x + size, y);

Graph.DrawLine(myPen, x + size, y - size, x + size, y + size);

}

}

private void DrawFractal(int x, int y, int size, int minSize)

{

int x1 = x - size;

int y1 = y - size;

int x2 = x - size;

int y2 = y + size;

int x3 = x + size;

int y3 = y - size;

int x4 = x + size;

int y4 = y + size;

DrawH(x, y, size);

size = size / 2;

if (size >= minSize)

{

DrawFractal(x1, y1, size, minSize);

DrawFractal(x2, y2, size, minSize);

DrawFractal(x3, y3, size, minSize);

DrawFractal(x4, y4, size, minSize);

}

}

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

{

int precision = 1;

int size = 1;

int.TryParse(FractPerc.Text, out precision);

int.TryParse(FractSize.Text, out size);

if (size < 1)

size = 1;

if (precision < 1)

precision = 1;

DrawFractal(width\_ / 2, height\_ / 2, size, precision);

}

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

{

using (Graphics Graph = PictBoxTask2.CreateGraphics())

{

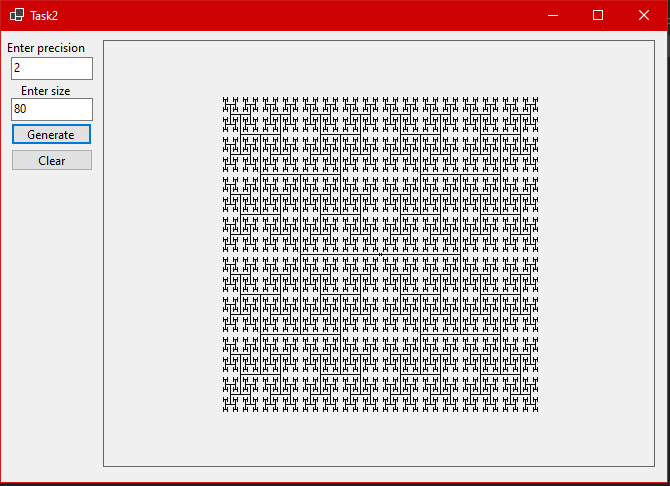
Graph.Clear(Color.LightGray);

}

}

}

}**Вывод программы:**

****

**Вывод:** в ходе выполнения лабораторной работы были получены навыки использования возможностей языка программирования C# в построении графических приложений.