1 задание.

Даны 4 точки A, B, C, D. Выясните, лежит ли точка A внутри угла BCD? Имеется в виду тот из углов BCD, который заметается лучом при повороте от CB к CD против часовой стрелки. (Решение этой задачи будет использовано в дальнейшем.)

Код:

using System;

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;

namespace Task1

{

public partial class Form1 : Form

{

int count = 0;//количество отмеченных точек

Bitmap bmp;

Graphics graph;

List<PointF> points;

public Form1()

{

InitializeComponent();

bmp = new Bitmap(pictureBox1.Width, pictureBox1.Height);

graph = Graphics.FromImage(bmp);

points = new List<PointF>();

}

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

{

if (count == 4)

{

if(IsClockWise(points[1], points[2], points[3])==-1)//В->C->D по часовой стрелке

{

if(IsClockWise(points[1], points[2], points[0])==-1&&

IsClockWise(points[0], points[2], points[3]) == -1)

MessageBox.Show("Точка A лежит внутри угла BCD");

else

MessageBox.Show("Точка A не лежит внутри угла BCD");

}

else if(IsClockWise(points[1], points[2], points[3]) == 1)//В->C->D против часовой стрелки

{

if (!(IsClockWise(points[1], points[2], points[0]) == 1 &&

IsClockWise(points[0], points[2], points[3]) == 1))

MessageBox.Show("Точка A лежит внутри угла BCD");

else

MessageBox.Show("Точка A не лежит внутри угла BCD");

}

}

}

private void pictureBox1\_MouseClick(object sender, MouseEventArgs e)

{

if (count < 4)

{

++count;

string word = null;

if (count == 1)

word = "A";

else if (count == 2)

word = "B";

else if (count == 3)

word = "C";

else if (count == 4)

word = "D";

graph.FillEllipse(Brushes.Black, e.X, e.Y, 3, 3);

graph.DrawString(word, new Font("Colibri", 7), Brushes.Blue, e.X - 10, e.Y - 10);

points.Add(new PointF(e.X, e.Y));

pictureBox1.Image = bmp;

}

if (count == 4)

{

graph.DrawLine(new Pen(Color.Black, 1), points[1], points[2]);

graph.DrawLine(new Pen(Color.Black, 1), points[2], points[3]);

pictureBox1.Image = bmp;

}

}

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

{

graph.Clear(pictureBox1.BackColor);

count = 0;

points.Clear();

pictureBox1.Image = bmp;

}

private int IsClockWise(PointF A, PointF B, PointF C)

{

if ((B.X - A.X) \* (C.Y - A.Y) - (B.Y - A.Y) \* (C.X - A.X) < 0) return 1;//против часовой стрелки

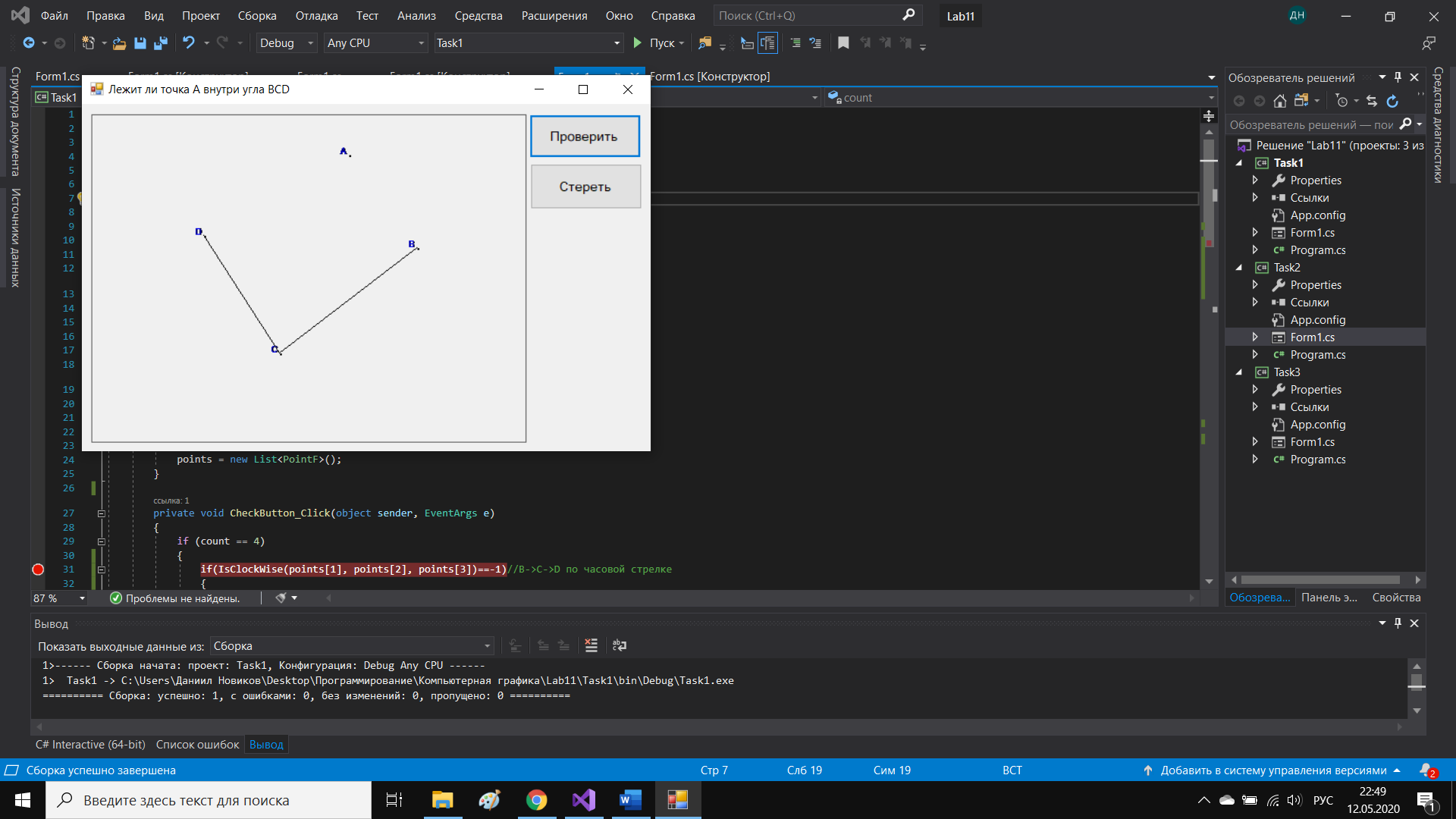
else return -1;//по часовой стрелке

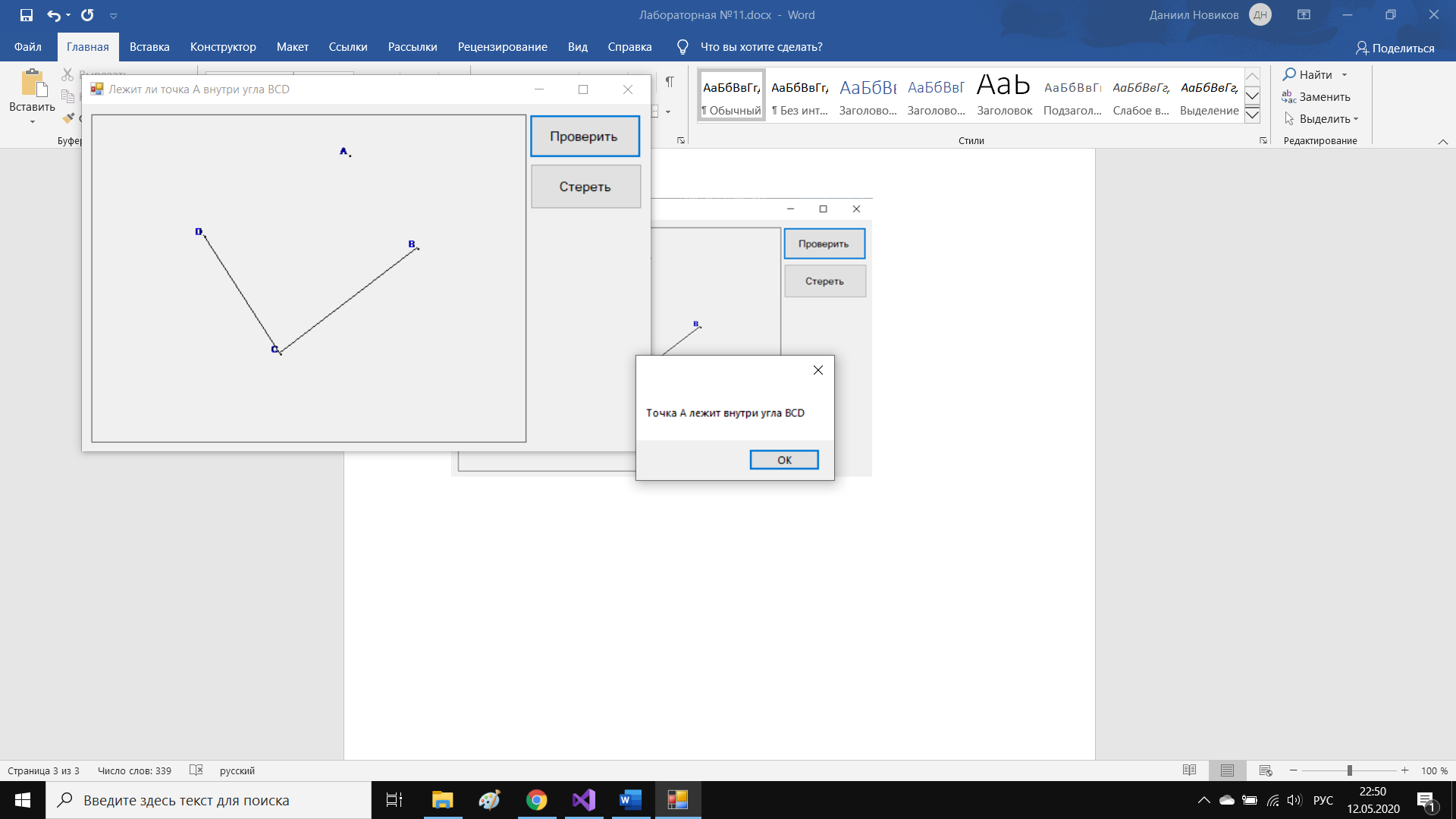
}

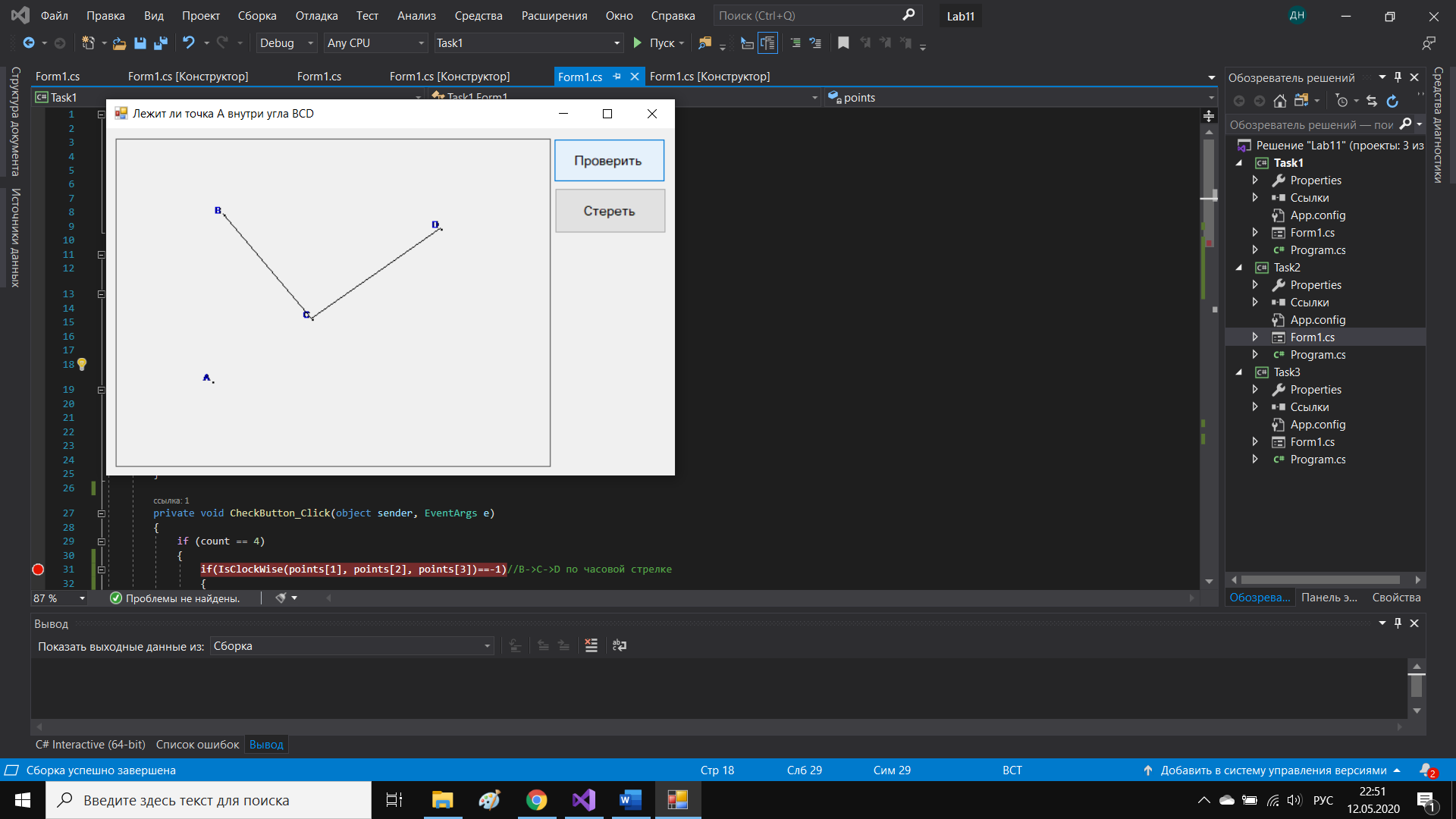
}

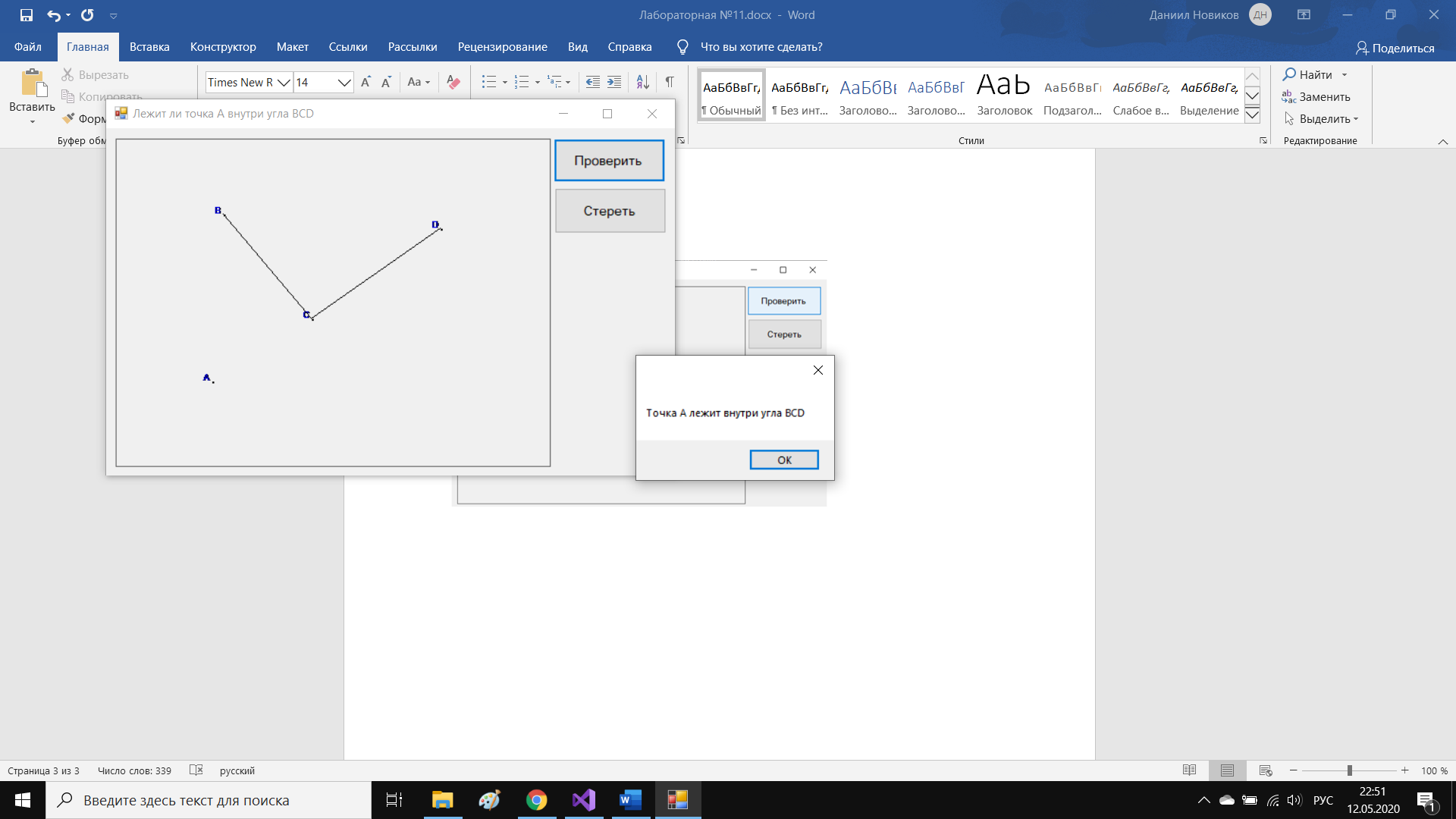
}

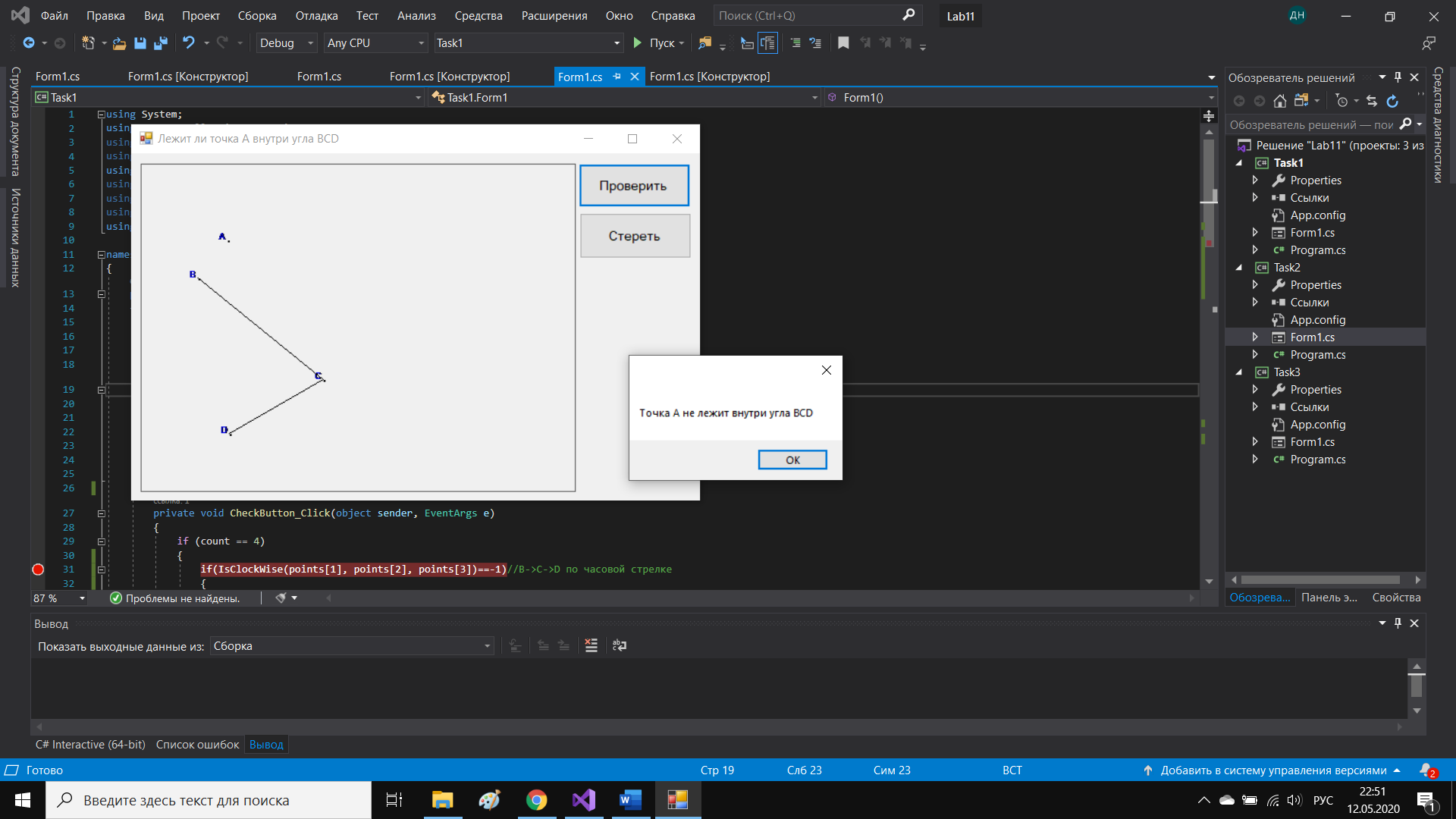
Результаты работы программы:











Задание 2.

Лежит ли данная точка A внутри данного многоугольника B1B2...Bn?

Код:

using System;

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;

namespace Task2

{

public partial class Form1 : Form

{

Bitmap bmp;

Graphics graph;

List<PointF> points;

bool checkDrawPolygon;

bool checkDrawPoint;

PointF a;

int countIntersect;//количество пересечений

public Form1()

{

InitializeComponent();

bmp = new Bitmap(pictureBox1.Width, pictureBox1.Height);

graph = Graphics.FromImage(bmp);

points = new List<PointF>();

}

private void pictureBox1\_MouseClick(object sender, MouseEventArgs e)

{

if (checkDrawPolygon)

{

graph.FillEllipse(Brushes.Black, e.X, e.Y, 3, 3);

points.Add(new PointF(e.X, e.Y));

pictureBox1.Image = bmp;

}

else if (checkDrawPoint)

{

graph.DrawPolygon(new Pen(Color.Black, 1), points.ToArray());

pictureBox1.Image = bmp;

a = new PointF(e.X, e.Y);

checkDrawPoint = false;

graph.FillEllipse(Brushes.Red, e.X, e.Y, 3, 3);

pictureBox1.Image = bmp;

}

}

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

{

checkDrawPolygon = true;

}

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

{

checkDrawPoint = true;

checkDrawPolygon = false;

}

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

{

checkDrawPolygon = false;

checkDrawPoint = false;

graph.Clear(pictureBox1.BackColor);

points.Clear();

pictureBox1.Image = bmp;

}

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

{

if (CheckForInside()) MessageBox.Show("Точка лежит внутри многоугольника");

else MessageBox.Show("Точка не лежит внутри многоугольника");

}

private bool CheckForInside()//проводим луч параллельно Ох и смотрим количество пересечений

{

for (int i = 0; i < points.Count(); i++)

{

PointF min = points[i];

PointF max = points[(i + 1) % points.Count()];

if (min.Y > max.Y)

{

PointF tmp = min;

min = max;

max = tmp;

}

if (max.Y <= a.Y || min.Y > a.Y)

continue;

int orient = IsClockWise(min, max, a);

if (orient == 0)//точка на ребре

return true;

if (orient < 0)

countIntersect++;

}

return (countIntersect % 2 == 0) ? false : true;

}

private int IsClockWise(PointF A, PointF B, PointF C)

{

if ((B.X - A.X) \* (C.Y - A.Y) - (B.Y - A.Y) \* (C.X - A.X) < 0) return 1;//по часовой стрелке

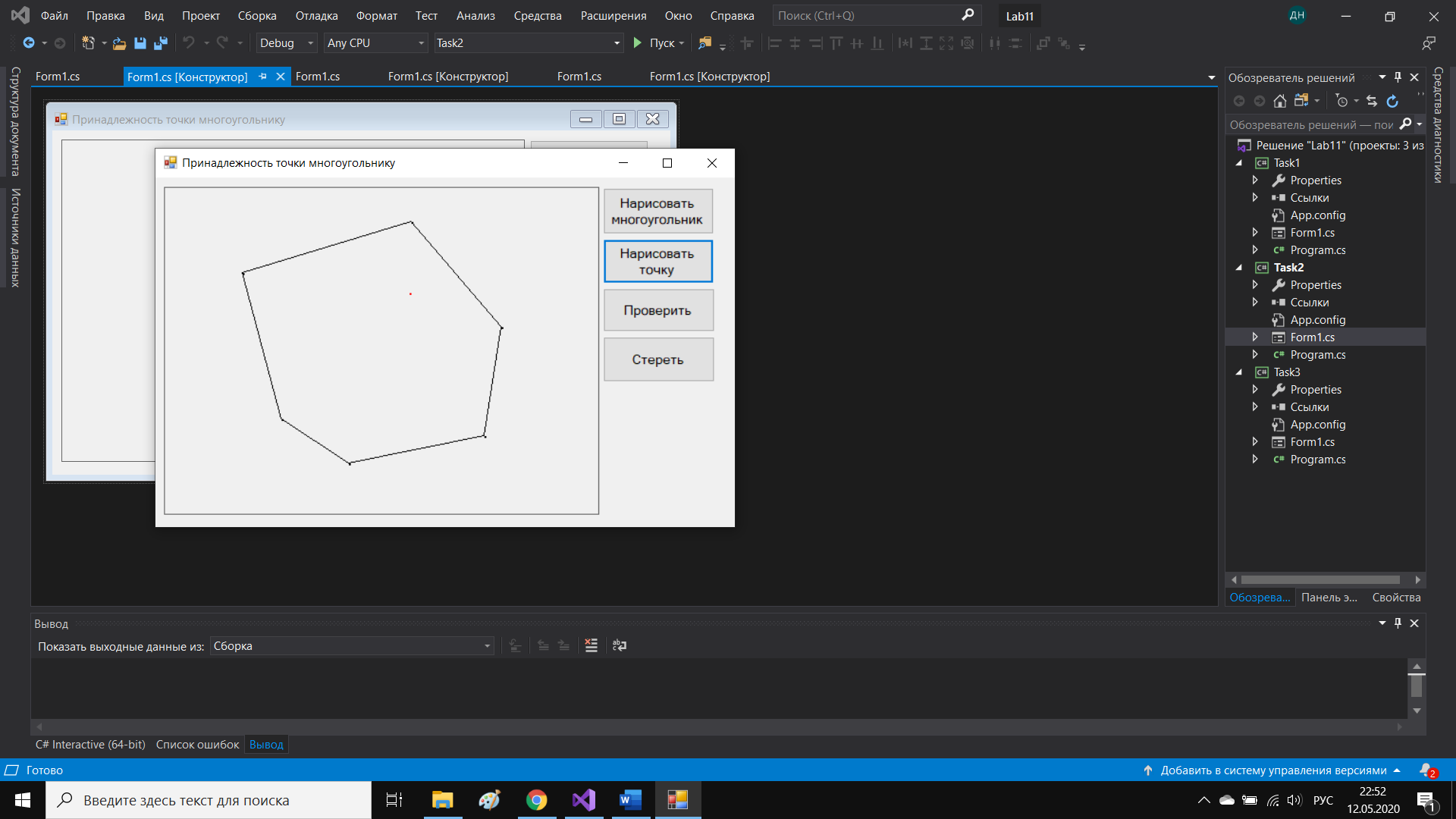
else return -1;//против часовой стрелки

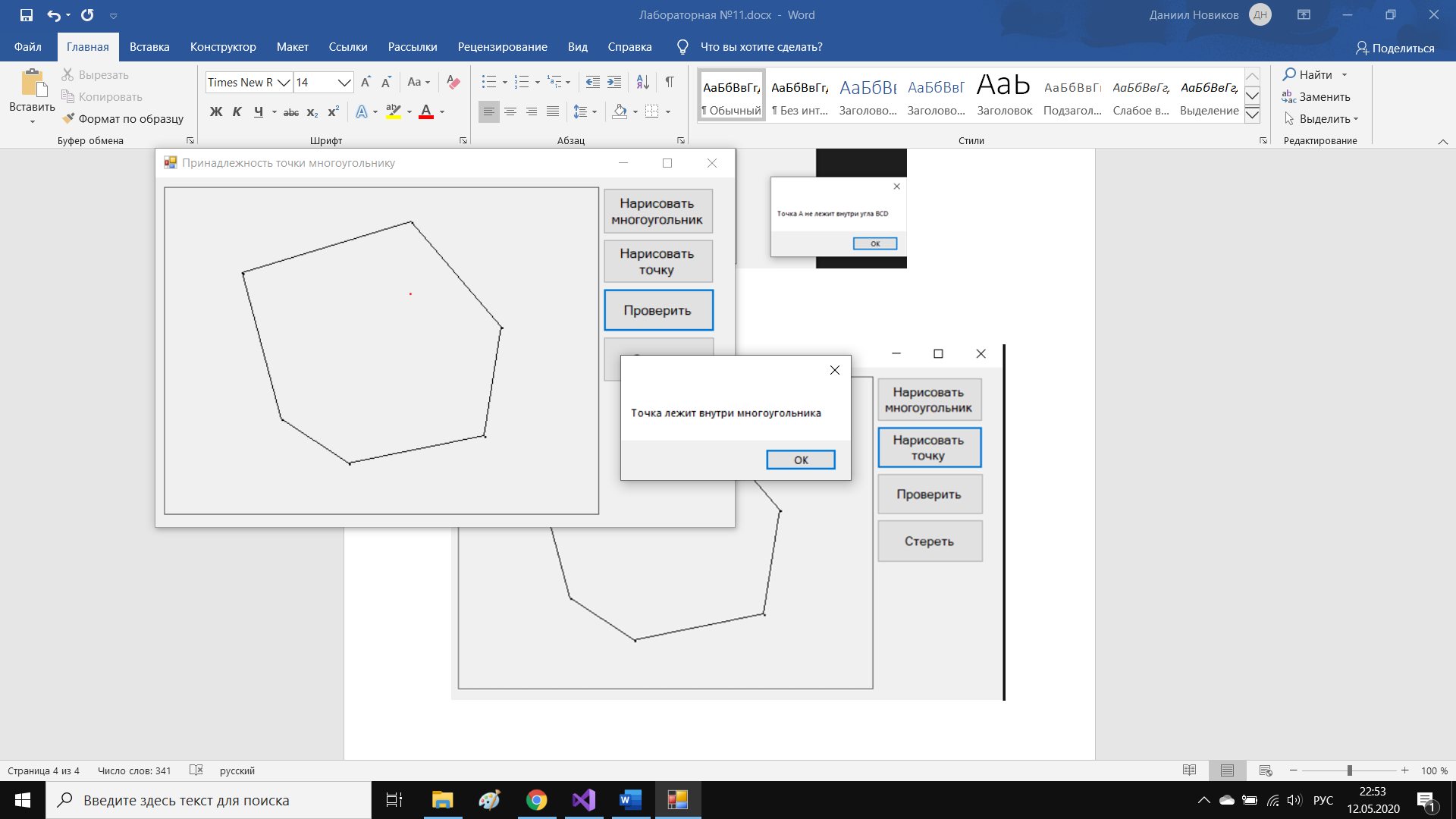
}

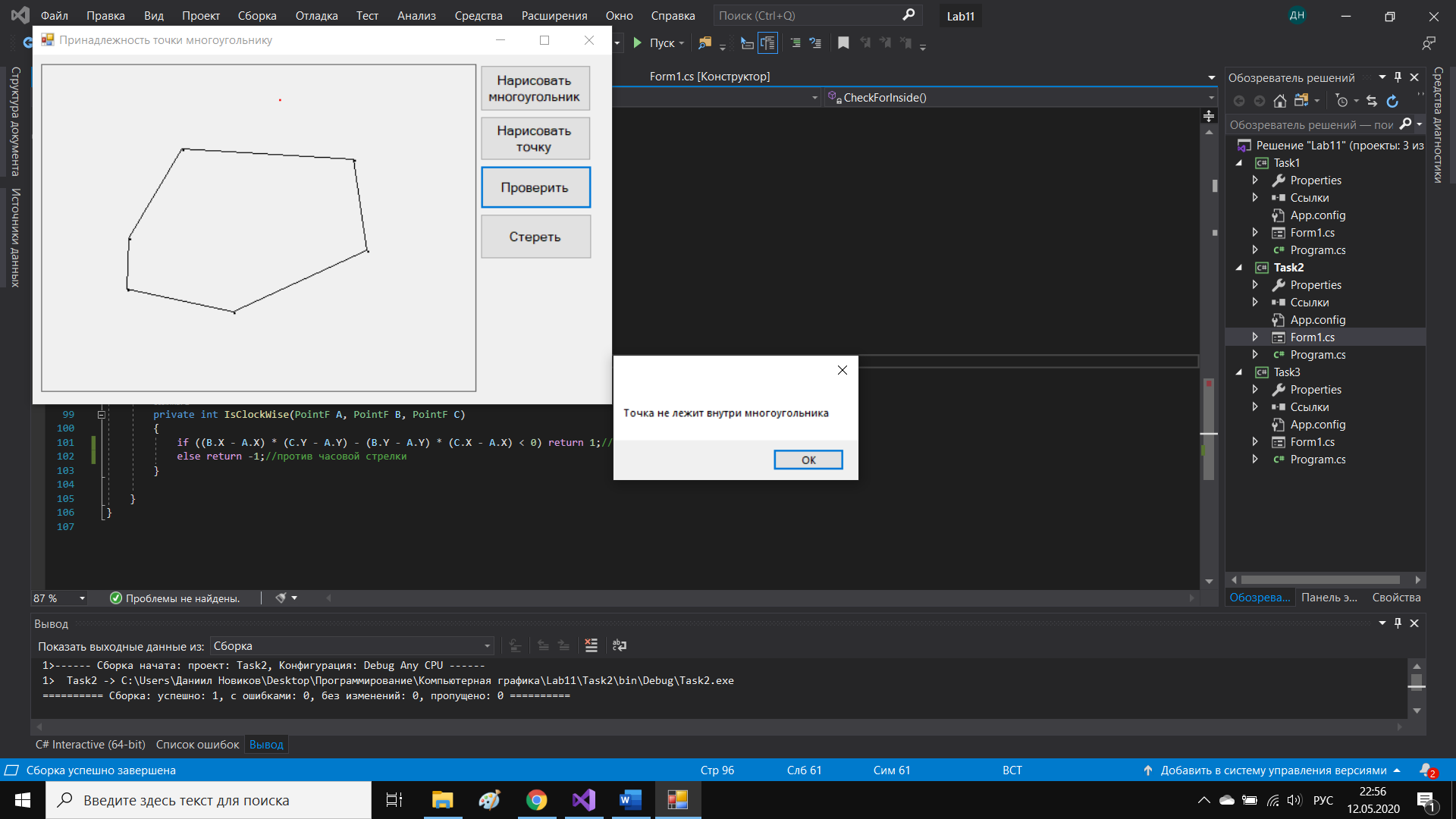
}

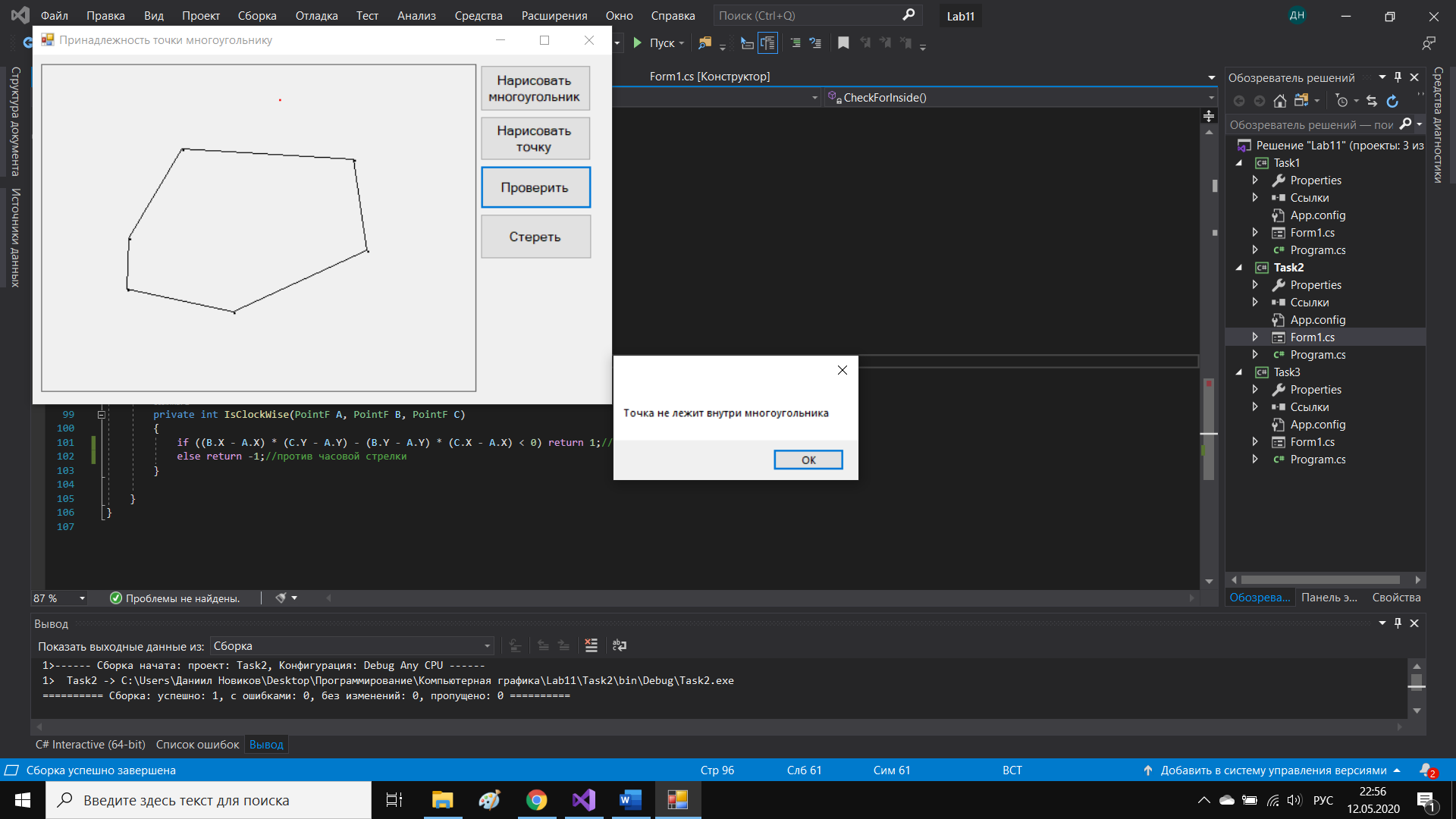
}

Результат работы программы:









3 задание.

Является ли данный многоугольник B1B2...Bn простым?

Код:

using System;

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;

namespace Task3

{

public partial class Form1 : Form

{

Bitmap bmp;

Graphics graph;

List<PointF> points;

public Form1()

{

InitializeComponent();

bmp = new Bitmap(pictureBox1.Width, pictureBox1.Height);

graph = Graphics.FromImage(bmp);

points = new List<PointF>();

}

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

{

graph.DrawPolygon(new Pen(Color.Black, 1), points.ToArray());

pictureBox1.Image = bmp;

if (CheckForSimplePolygon())

MessageBox.Show("Многоугольник простой");

else

MessageBox.Show("Многоугольник не является простым");

}

/// <summary>

/// Проверка на простоту многоугольника

/// </summary>

/// <returns></returns>

private bool CheckForSimplePolygon()

{

int n = points.Count();

int count = 0;//количество сторон, которые уже проверили с исходной

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

{

for (int j = (i + 2) % n; count < n - 3; j = (j + 1) % n, count += 1)

{

if (CheckForIntersect(points[i], points[i + 1], points[j], points[(j + 1)%n]))

return false;//появилось перечесение

}

count = 0;

}

return true;

}

/// <summary>

/// проверка, находится ли точка B между A и C

/// </summary>

/// <param name="A"></param>

/// <param name="B"></param>

/// <param name="C"></param>

/// <returns></returns>

private bool CheckForStayinSegment(PointF A, PointF B, PointF C)

{

return (B.X <= Math.Max(A.X, C.X) && B.X >= Math.Min(A.X, C.X) &&

B.Y <= Math.Max(A.Y, C.Y) && B.Y >= Math.Min(A.Y, C.Y)) ? true : false;

}

private bool CheckForIntersect(PointF A, PointF B, PointF C, PointF D)//АВ-первый отрезок, CD-второй отрезок

{

int orient1 = Orientation(A, B, C);

int orient2 = Orientation(A, B, D);

int orient3 = Orientation(C, D, A);

int orient4 = Orientation(C, D, B);

if (orient1 != orient2 && orient3 != orient4)

return true;

if ((orient1 == 0 && CheckForStayinSegment(A, C, B)) ||

(orient2 == 0 && CheckForStayinSegment(A, D, B)) ||

(orient3 == 0 && CheckForStayinSegment(C, A, D)) ||

(orient4 == 0 && CheckForStayinSegment(C, B, D))) return true;

else return false;

}

private int Orientation(PointF A, PointF B, PointF C)

{

if ((B.X - A.X) \* (C.Y - A.Y) - (B.Y - A.Y) \* (C.X - A.X) < 0) return 1;//по часовой стрелке

else return -1;//против часовой стрелки

}

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

{

graph.Clear(pictureBox1.BackColor);

points.Clear();

pictureBox1.Image = bmp;

}

private void pictureBox1\_MouseClick(object sender, MouseEventArgs e)

{

graph.FillEllipse(Brushes.Black, e.X, e.Y, 3, 3);

points.Add(new PointF(e.X, e.Y));

pictureBox1.Image = bmp;

}

}

}

Результат работы программы:

