Задание №1

Условие:

Разработайте программу анимации летающего бумеранга.

Исходный код программы:

using System;

using System.Drawing;

using System.Windows.Forms;

namespace Boomerang

{

public partial class MainForm : Form

{

PointF posB = new PointF();

float R = 300;

PointF Center = new PointF(500, 500);

float Ang = 0f;

float RotAng = 0f;

public MainForm()

{

InitializeComponent();

var t = new Timer();

t.Interval = 30;

t.Enabled = true;

t.Tick += (s, o) => { MoveB(); };

}

private void MoveB()

{

RotAng += 30f;

Ang += 0.1f;

var x = Center.X + R \* (float)Math.Cos(Ang);

var y = Center.Y + R \* (float)Math.Sin(Ang);

posB = new PointF(x, y);

Invalidate();

}

protected override void OnPaint(PaintEventArgs e)

{

e.Graphics.TranslateTransform(posB.X / 2, posB.Y / 2);

e.Graphics.RotateTransform(RotAng);

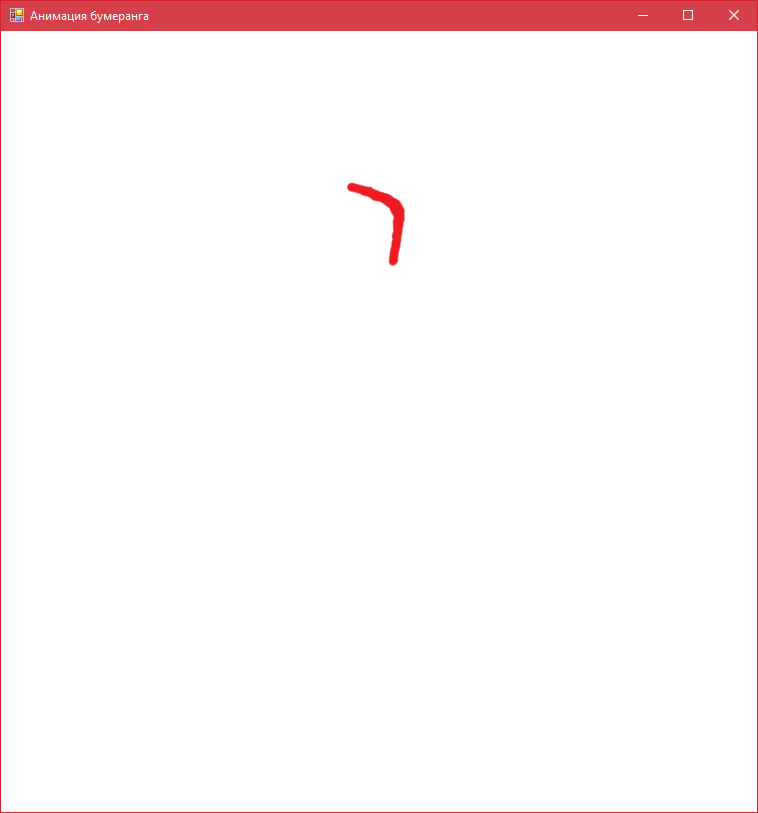
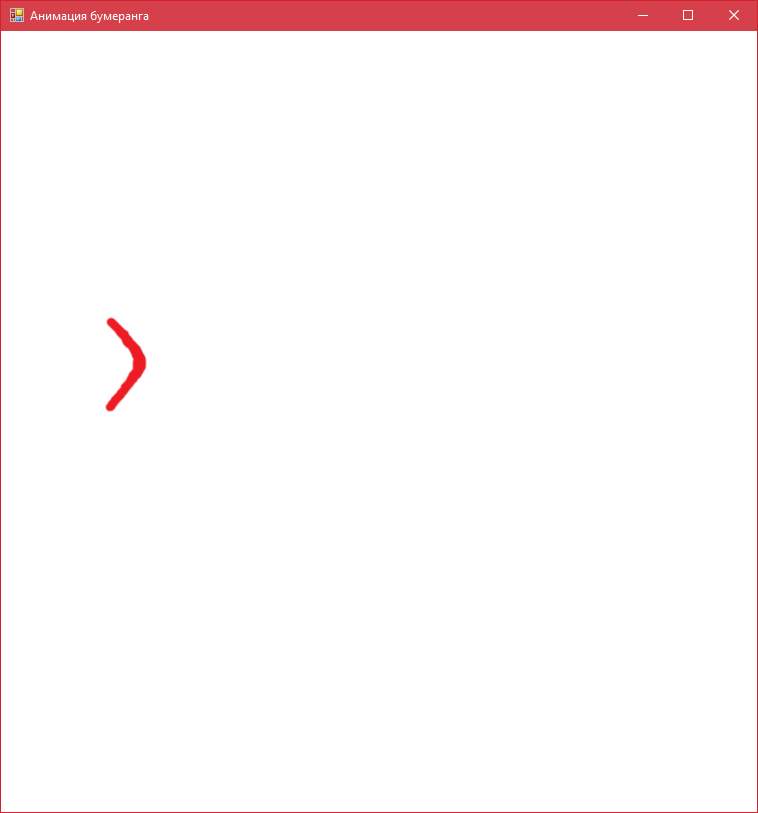
e.Graphics.DrawImage(Res.bu, new PointF(-Res.bu.Width / 2, -Res.bu.Height / 2));

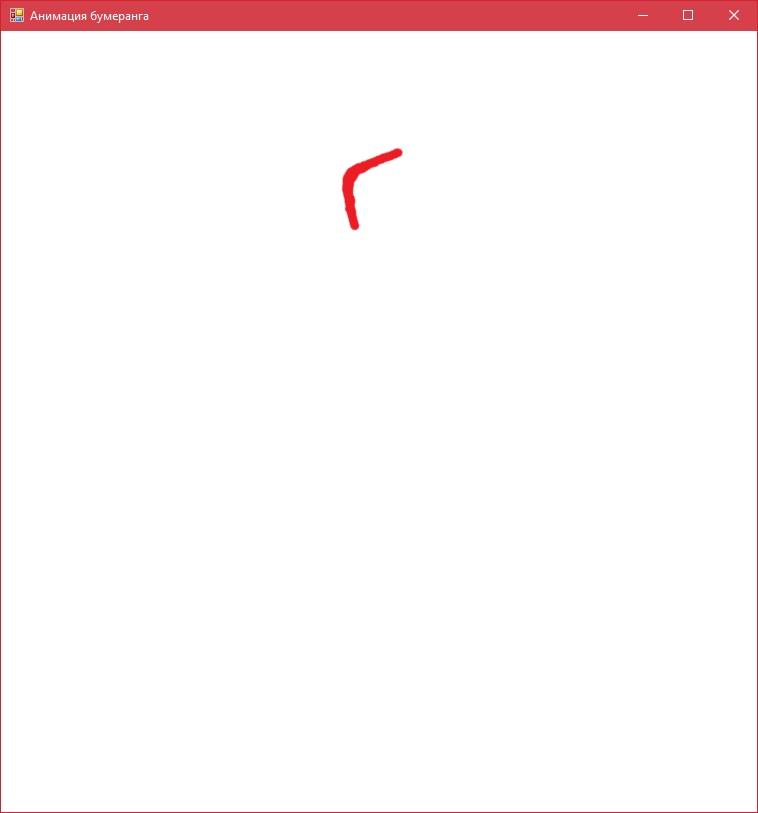
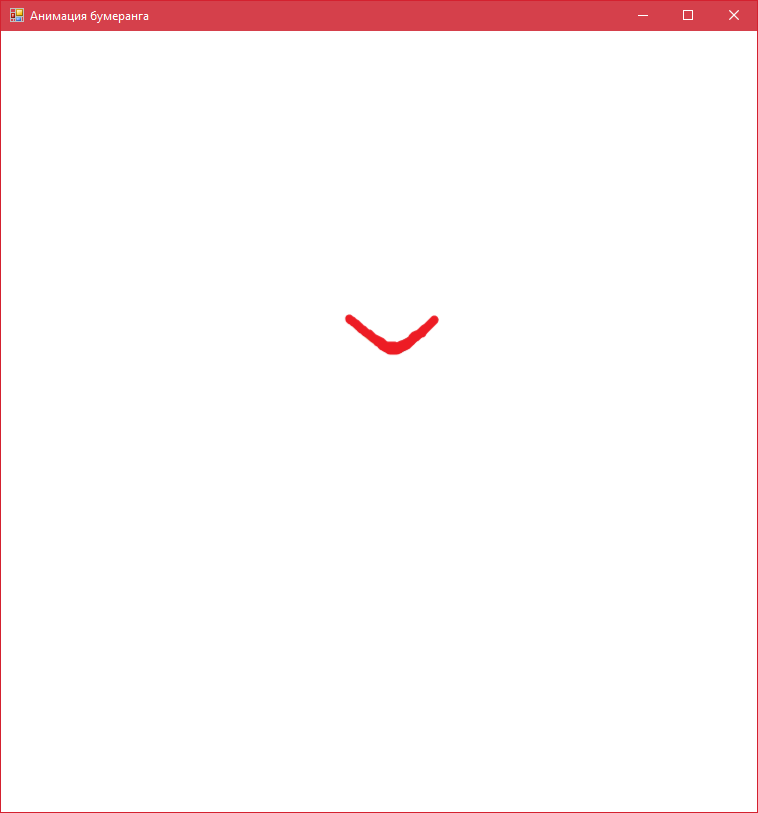
}

}

}

Скриншоты программы:

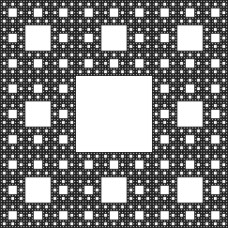




Задание №2

Условие:

Постройте ковер Серпинского. Каждый вызов рекурсивной функции происходит по таймеру.



Исходный код программы:

using System;

using System.Drawing;

using System.Windows.Forms;

namespace WindowsFormsGraphicsExample

{

public partial class Form1 : Form

{

private const int Level = 5;

private int \_width;

private int \_height;

private Bitmap \_fractal;

private Graphics \_graph;

public Form1()

{

InitializeComponent();

\_width = FractalPictureBox.Width;

\_height = FractalPictureBox.Height;

}

private PointF MidPoint(PointF p1, PointF p2)

{

return new PointF((p1.X + p2.X) / 2f, (p1.Y + p2.Y) / 2f);

}

private void DrawCarpet(int level, RectangleF carpet)

{

if (level == 0)

\_graph.FillRectangle(Brushes.OrangeRed, carpet);

else

{

var width = carpet.Width / 3f;

var height = carpet.Height / 3f;

var x1 = carpet.Left;

var x2 = x1 + width;

var x3 = x1 + 2f \* width;

var y1 = carpet.Top;

var y2 = y1 + height;

var y3 = y1 + 2f \* height;

DrawCarpet(level - 1, new RectangleF(x1, y1, width, height));

DrawCarpet(level - 1, new RectangleF(x2, y1, width, height));

DrawCarpet(level - 1, new RectangleF(x3, y1, width, height));

DrawCarpet(level - 1, new RectangleF(x1, y2, width, height));

DrawCarpet(level - 1, new RectangleF(x3, y2, width, height));

DrawCarpet(level - 1, new RectangleF(x1, y3, width, height));

DrawCarpet(level - 1, new RectangleF(x2, y3, width, height));

DrawCarpet(level - 1, new RectangleF(x3, y3, width, height));

}

}

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

{

\_fractal = new Bitmap(\_width, \_height);

\_graph = Graphics.FromImage(\_fractal);

RectangleF carpet = new RectangleF(0, 0, \_width, \_height);

DrawCarpet(Level, carpet);

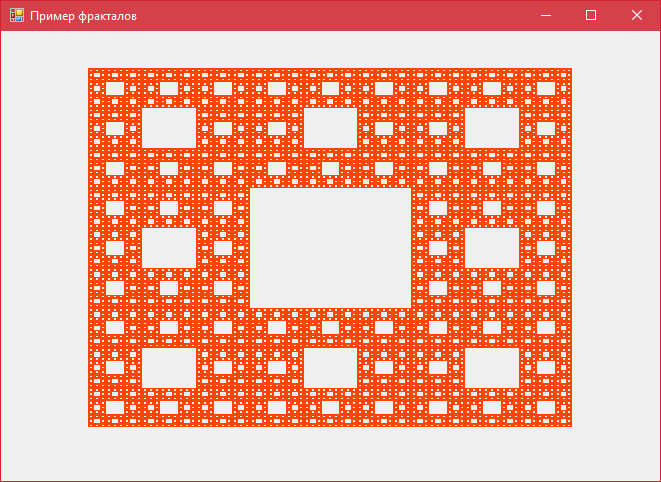
FractalPictureBox.BackgroundImage = \_fractal;

}

}

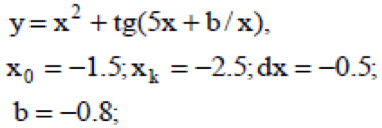
}

Скриншоты программы:



Задание №3

Условие:

Изучить возможности построения графиков с помощью элемента управления Сhart. Написать и отладить программу построения на экране графика функции:

Исходный код программы:

using System;

using System.Windows.Forms;

namespace WindowsFormsCoolGraph

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void GraphInitializer()

{

double Xmin = Convert.ToDouble(textBoxXmin.Text);

double Xmax = Convert.ToDouble(textBoxXmax.Text);

double Step = Convert.ToDouble(textBoxStep.Text);

double b = -0.8;

int count = (int)Math.Ceiling((Xmax - Xmin) / Step) + 1;

double[] x = new double[count];

double[] y1 = new double[count];

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

{

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

y1[i] = Math.Pow(i, 2) + Math.Tan((5 \* i) + (b / i));

}

chart1.ChartAreas[0].AxisX.Minimum = Xmin; chart1.ChartAreas[0].AxisX.Maximum = Xmax;

chart1.ChartAreas[0].AxisX.MajorGrid.Interval = Step;

chart1.Series[0].Points.DataBindXY(x, y1);

}

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

{

}

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

{

GraphInitializer();

}

private void chart1\_Click(object sender, EventArgs e)

{

}

}

}

Скриншоты программы:

