Батраков ИС224

using System.Drawing.Drawing2D;

namespace WinFormsApp1

{

public partial class Form1 : Form

{

private bool dragging = false; // Флаг, указывающий, перетаскивается ли форма

private Point dragCursorPoint; // Точка курсора

private Point dragFormPoint; // Точка формы

public Form1()

{

InitializeComponent();

this.MouseDown += new MouseEventHandler(Form1\_MouseDown);

this.MouseMove += new MouseEventHandler(Form1\_MouseMove);

this.MouseUp += new MouseEventHandler(Form1\_MouseUp);

this.ClientSize = new Size(400, 400);

this.Text = "Фиолетовая звезда";

SetFormToStar();

this.Paint += new PaintEventHandler(MainForm\_Paint);

}

private void SetFormToStar()

{

// Calculate points for a star

int n = 5; // Number of points

double r = Math.Min(ClientSize.Width, ClientSize.Height) / 3; // Radius of the star

double x0 = ClientSize.Width / 2; // Center x

double y0 = ClientSize.Height / 2; // Center y

double a = 0;

double da = Math.PI / n;

PointF[] starPoints = new PointF[2 \* n];

for (int i = 0; i < 2 \* n; i++)

{

double angle = a + (i % 2 == 0 ? 0 : da / 2); // Alternate angles for outer/inner points

double l = (i % 2 == 0) ? r : r / 2; // Use smaller radius for inner points

starPoints[i] = new PointF((float)(x0 + l \* Math.Cos(angle)), (float)(y0 + l \* Math.Sin(angle)));

a += da;

}

// Create the star path

GraphicsPath path = new GraphicsPath();

path.AddPolygon(starPoints);

// Set the form's region

this.Region = new Region(path);

}

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

{

Graphics g = e.Graphics;

int n = 10;

// Adjust these values for scaling

double R = 500, r = 500; // Increase radii

double alpha = 45.0;

double x0 = 200, y0 = 300; // Shift the center away from the origin

PointF[] points = new PointF[2 \* n + 1];

double a = alpha, da = Math.PI / n, l;

for (int k = 0; k < 2 \* n + 1; k++)

{

l = k % 2 == 0 ? r : R;

points[k] = new PointF((float)(x0 + l \* Math.Cos(a)), (float)(y0 + l \* Math.Sin(a)));

a += da;

}

g.FillPolygon(new System.Drawing.SolidBrush(Color.Violet), points);

}

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

{

}

private void Form1\_MouseDown(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

dragging = true;

dragCursorPoint = Cursor.Position;

dragFormPoint = this.Location;

}

}

private void Form1\_MouseMove(object sender, MouseEventArgs e)

{

if (dragging)

{

Point dif = Point.Subtract(Cursor.Position, new Size(dragCursorPoint));

this.Location = Point.Add(dragFormPoint, new Size(dif));

}

}

private void Form1\_MouseUp(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

dragging = false;

}

}

}

}

