Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

Кафедра прикладной математики и кибернетики

Отчет по лабораторной работе

по дисциплине «Визуальное программирование»

Лабораторная работа № 5

«Перетаскивание графических объектов с помощью мыши»

Выполнил: студент 2 курса группы ИП-811 Мироненко К. А

Проверил: доцент кафедры ПМиК Ситняковская Е. И.

**Оглавление**

[1. Постановка задачи 3](#_Toc42200259)

[2. Описание работы 4](#_Toc42200260)

[3. Примеры работы программы 5](#_Toc42200261)

[*Приложение* Листинг 8](#_Toc42200262)

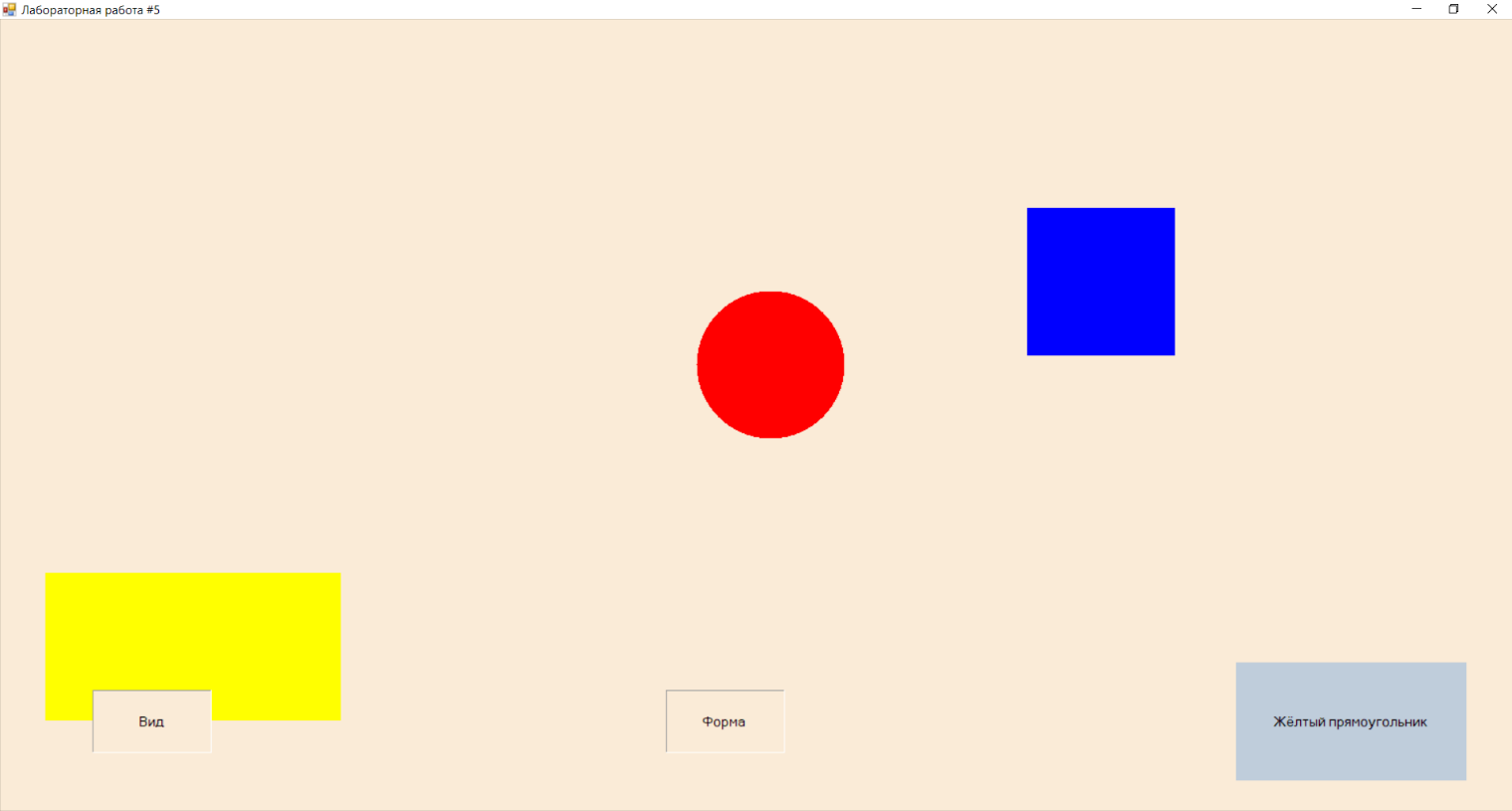
# Постановка задачи

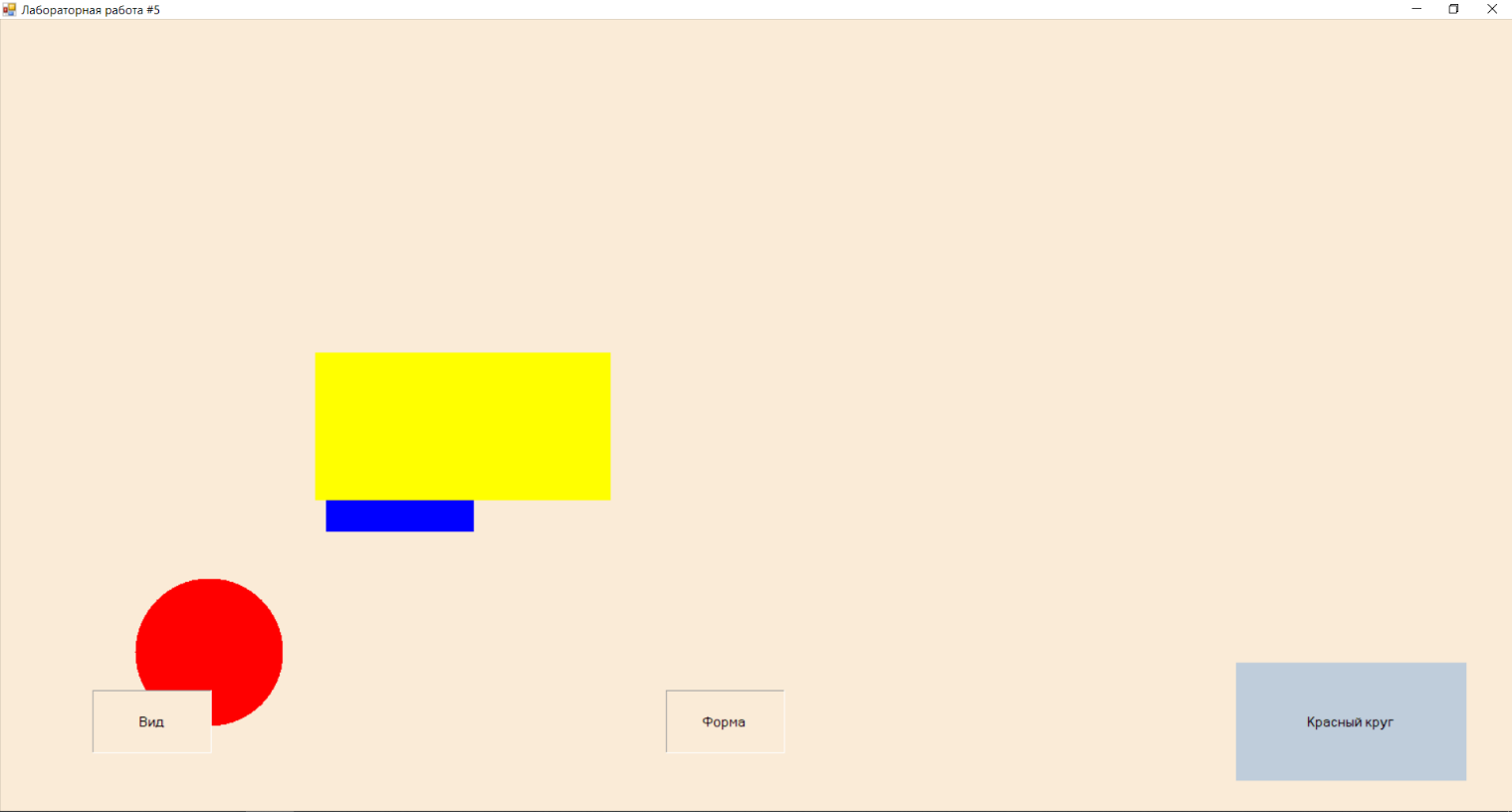
1. На форме, в границах PictureBox, должны быть созданы три графических объекта: квадрат, эллипс и прямоугольник.
2. Реализована возможность перетаскивания этих объектов по форме.
3. Должно быть выполнено следующее условие: если один из объектов, будет перенесён на Label «Вид», то в третьем Lable «Информация», должны появиться данные о том какого цвета и что это за фигура.
4. При перетаскивании окружности или квадрата на Label «Форма», объект должен меняться на противоположный (т.е. если перенесли квадрат, то он становится окружностью, а окружность, в свою очередь, становится квадратом и наоборот).

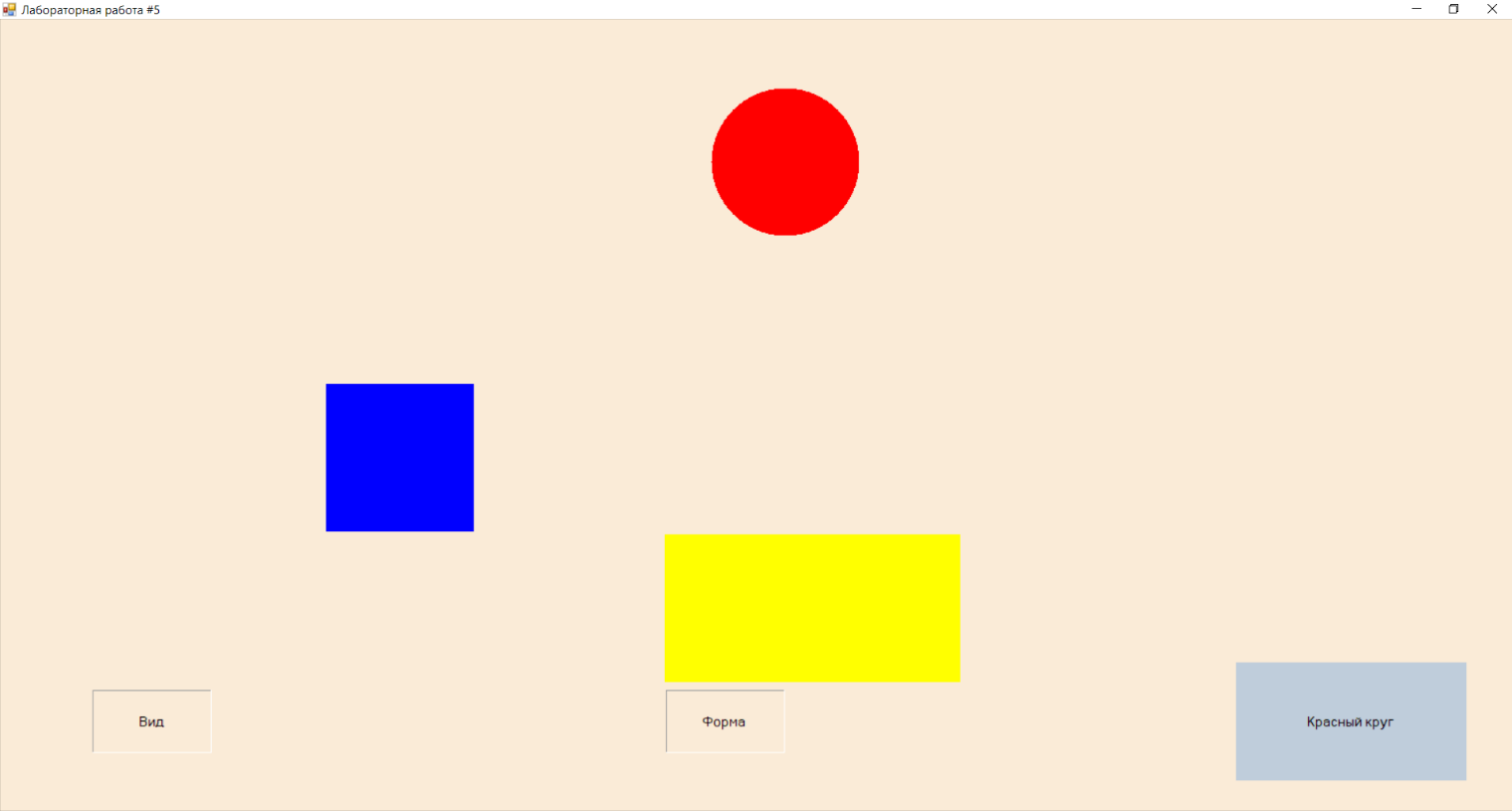
# Описание работы

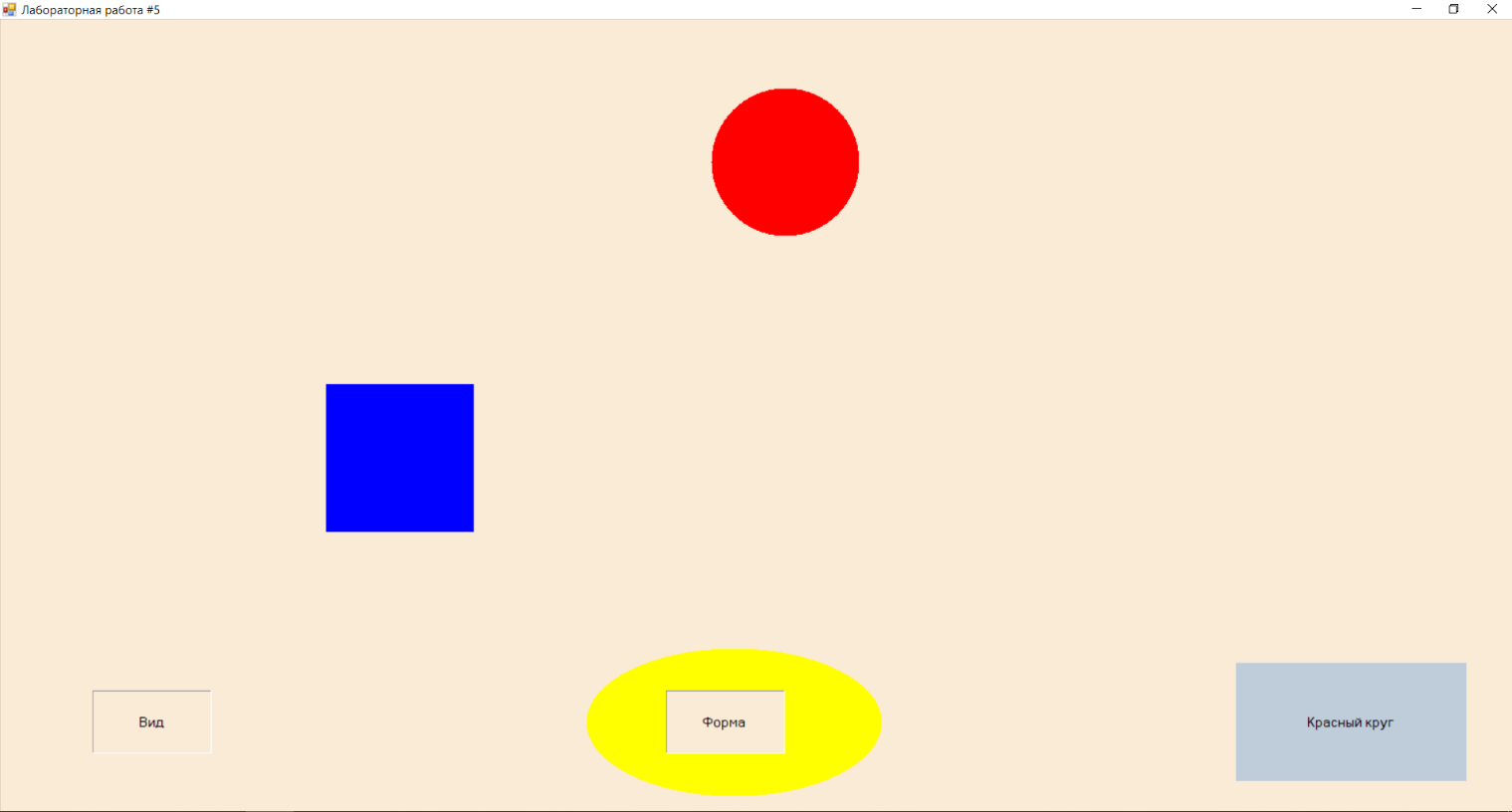
Работа приложения соответствует описанию в методичке, за исключением именования и свойств некоторых элементов.

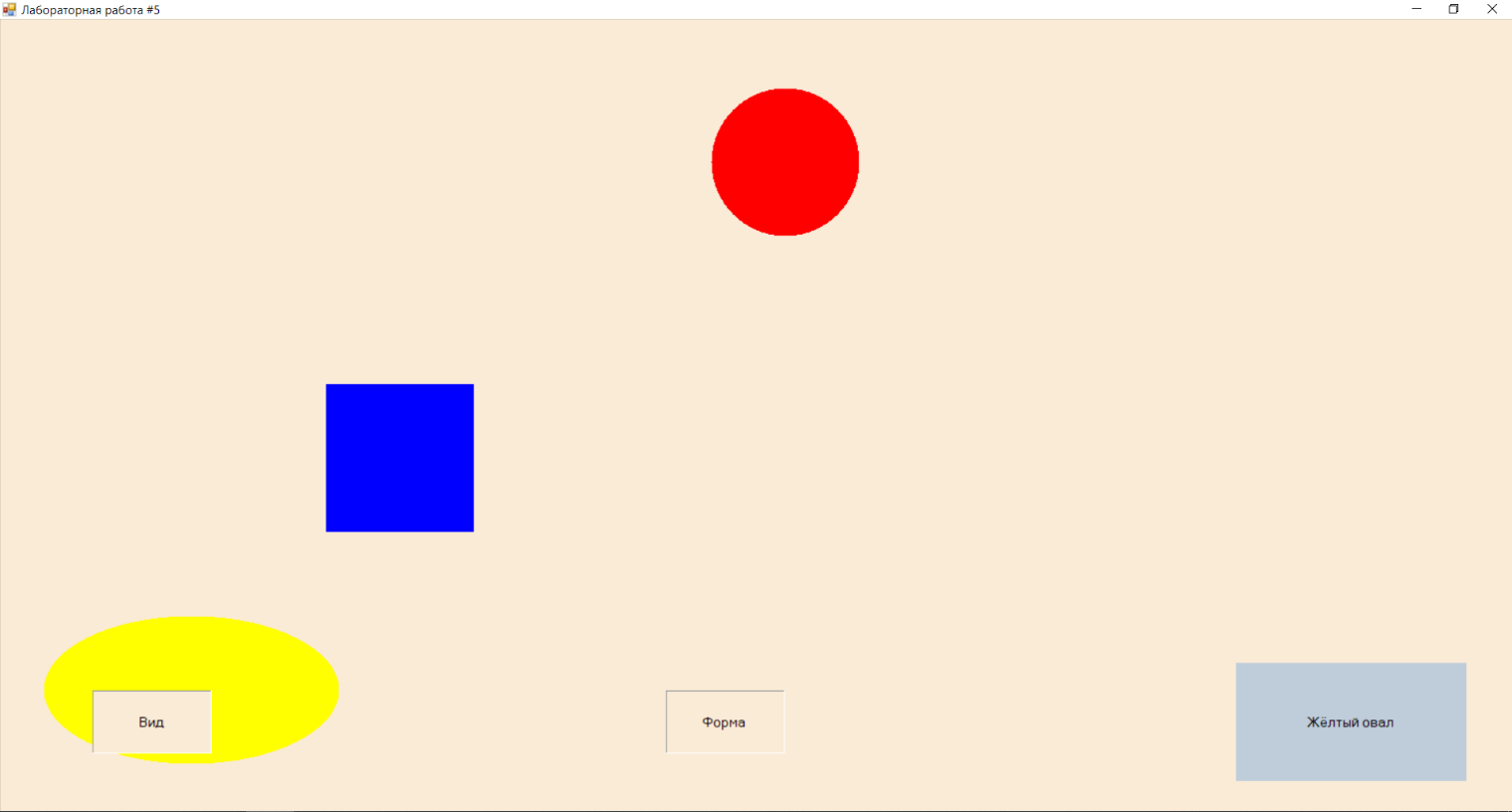
# Примеры работы программы

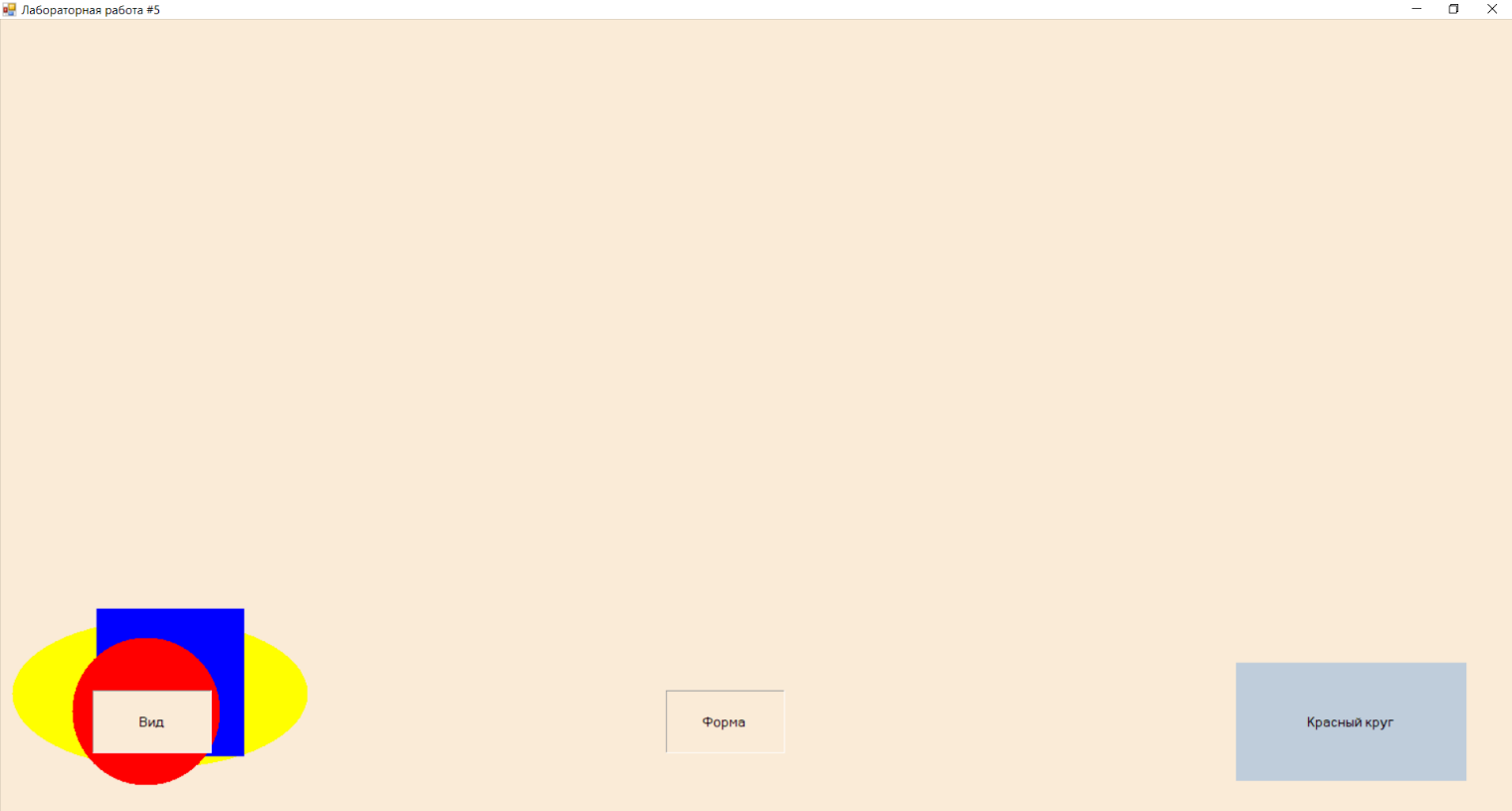












# *Приложение* Листинг

**Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace laboratoryWork\_5

{

static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new FormMain());

}

}

}

**FormMain.cs**

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 laboratoryWork\_5

{

public partial class FormMain : Form

{

private Rectangle rectangle, circle, square;

private bool bool\_rectangle, bool\_circle, bool\_square;

private bool bool\_rounded\_rectangle, bool\_rounded\_circle, bool\_rounded\_square;

private int rectangleZ, circleZ, squareZ;

private bool dragging;

private int oldX, oldY;

public FormMain()

{

InitializeComponent();

rectangle = new Rectangle(10, 10, 300, 150);

rectangleZ = 1;

bool\_rounded\_rectangle = false;

circle = new Rectangle(420, 10, 150, 150);

circleZ = 2;

bool\_rounded\_circle = true;

square = new Rectangle(630, 10, 150, 150);

squareZ = 3;

bool\_rounded\_square = false;

dragging = false;

}

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

{

if (rectangleZ < circleZ && rectangleZ < squareZ)

{

if (bool\_rounded\_rectangle) e.Graphics.FillEllipse(Brushes.Yellow, rectangle);

else e.Graphics.FillRectangle(Brushes.Yellow, rectangle);

if (circleZ < squareZ)

{

if (bool\_rounded\_circle) e.Graphics.FillEllipse(Brushes.Red, circle);

else e.Graphics.FillRectangle(Brushes.Red, circle);

if (bool\_rounded\_square) e.Graphics.FillEllipse(Brushes.Blue, square);

else e.Graphics.FillRectangle(Brushes.Blue, square);

}

else

{

if (bool\_rounded\_square) e.Graphics.FillEllipse(Brushes.Blue, square);

else e.Graphics.FillRectangle(Brushes.Blue, square);

if (bool\_rounded\_circle) e.Graphics.FillEllipse(Brushes.Red, circle);

else e.Graphics.FillRectangle(Brushes.Red, circle);

}

}

else if (circleZ < rectangleZ && circleZ < squareZ)

{

if (bool\_rounded\_circle) e.Graphics.FillEllipse(Brushes.Red, circle);

else e.Graphics.FillRectangle(Brushes.Red, circle);

if (rectangleZ < squareZ)

{

if (bool\_rounded\_rectangle) e.Graphics.FillEllipse(Brushes.Yellow, rectangle);

else e.Graphics.FillRectangle(Brushes.Yellow, rectangle);

if (bool\_rounded\_square) e.Graphics.FillEllipse(Brushes.Blue, square);

else e.Graphics.FillRectangle(Brushes.Blue, square);

}

else

{

if (bool\_rounded\_square) e.Graphics.FillEllipse(Brushes.Blue, square);

else e.Graphics.FillRectangle(Brushes.Blue, square);

if (bool\_rounded\_rectangle) e.Graphics.FillEllipse(Brushes.Yellow, rectangle);

else e.Graphics.FillRectangle(Brushes.Yellow, rectangle);

}

}

else

{

if (bool\_rounded\_square) e.Graphics.FillEllipse(Brushes.Blue, square);

else e.Graphics.FillRectangle(Brushes.Blue, square);

if (circleZ < rectangleZ)

{

if (bool\_rounded\_circle) e.Graphics.FillEllipse(Brushes.Red, circle);

else e.Graphics.FillRectangle(Brushes.Red, circle);

if (bool\_rounded\_rectangle) e.Graphics.FillEllipse(Brushes.Yellow, rectangle);

else e.Graphics.FillRectangle(Brushes.Yellow, rectangle);

}

else

{

if (bool\_rounded\_rectangle) e.Graphics.FillEllipse(Brushes.Yellow, rectangle);

else e.Graphics.FillRectangle(Brushes.Yellow, rectangle);

if (bool\_rounded\_circle) e.Graphics.FillEllipse(Brushes.Red, circle);

else e.Graphics.FillRectangle(Brushes.Red, circle);

}

}

}

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

{

oldX = e.X;

oldY = e.Y;

int tmp = 0;

if (oldX >= rectangle.X &&

oldX <= rectangle.X + rectangle.Width &&

oldY >= rectangle.Y &&

oldY <= rectangle.Y + rectangle.Height)

{

tmp = rectangleZ;

}

if (oldX >= circle.X &&

oldX <= circle.X + circle.Width &&

oldY >= circle.Y &&

oldY <= circle.Y + circle.Height)

{

if (tmp < circleZ)

{

tmp = circleZ;

}

}

if (oldX >= square.X &&

oldX <= square.X + square.Width &&

oldY >= square.Y &&

oldY <= square.Y + square.Height)

{

if (tmp < squareZ)

{

tmp = squareZ;

}

}

if (tmp != 0)

{

dragging = true;

if (tmp == rectangleZ)

{

bool\_rectangle = true;

rectangleZ = 3;

if (circleZ > squareZ)

{

circleZ = 2;

squareZ = 1;

}

else

{

squareZ = 2;

circleZ = 1;

}

}

else if (tmp == circleZ)

{

bool\_circle = true;

circleZ = 3;

if (rectangleZ > squareZ)

{

rectangleZ = 2;

squareZ = 1;

}

else

{

squareZ = 2;

rectangleZ = 1;

}

}

else

{

bool\_square = true;

squareZ = 3;

if (circleZ > rectangleZ)

{

circleZ = 2;

rectangleZ = 1;

}

else

{

rectangleZ = 2;

circleZ = 1;

}

}

}

}

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

{

if (dragging)

{

if (bool\_rectangle)

{

rectangle.X = e.X - (oldX - rectangle.X);

rectangle.Y = e.Y - (oldY - rectangle.Y);

if (

((labelView.Location.X < rectangle.X + rectangle.Width) &&

(labelView.Location.X > rectangle.X) &&

(labelView.Location.Y < rectangle.Y + rectangle.Height) &&

(labelView.Location.Y > rectangle.Y)

) ||

((labelView.Location.X + labelView.Width < rectangle.X + rectangle.Width) &&

(labelView.Location.X + labelView.Width > rectangle.X) &&

(labelView.Location.Y < rectangle.Y + rectangle.Height) &&

(labelView.Location.Y > rectangle.Y)

)

)

{

labelInfo.Text = "Жёлтый ";

if (bool\_rounded\_rectangle)

labelInfo.Text += "овал";

else

labelInfo.Text += "прямоугольник";

}

}

else if (bool\_circle)

{

circle.X = e.X - (oldX - circle.X);

circle.Y = e.Y - (oldY - circle.Y);

if (

((labelView.Location.X < circle.X + circle.Width) &&

(labelView.Location.X > circle.X) &&

(labelView.Location.Y < circle.Y + circle.Height) &&

(labelView.Location.Y > circle.Y)

) ||

((labelView.Location.X + labelView.Width < circle.X + circle.Width) &&

(labelView.Location.X + labelView.Width > circle.X) &&

(labelView.Location.Y < circle.Y + circle.Height) &&

(labelView.Location.Y > circle.Y)

)

)

{

labelInfo.Text = "Красный ";

if (bool\_rounded\_circle)

labelInfo.Text += "круг";

else

labelInfo.Text += "квадрат";

}

}

else

{

square.X = e.X - (oldX - square.X);

square.Y = e.Y - (oldY - square.Y);

if (

((labelView.Location.X < square.X + square.Width) &&

(labelView.Location.X > square.X) &&

(labelView.Location.Y < square.Y + square.Height) &&

(labelView.Location.Y > square.Y)

) ||

((labelView.Location.X + labelView.Width < square.X + square.Width) &&

(labelView.Location.X + labelView.Width > square.X) &&

(labelView.Location.Y < square.Y + square.Height) &&

(labelView.Location.Y > square.Y)

)

)

{

labelInfo.Text = "Синий ";

if (bool\_rounded\_square)

labelInfo.Text += "круг";

else

labelInfo.Text += "квадрат";

}

}

oldX = e.X;

oldY = e.Y;

}

pictureBox.Invalidate();

}

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

{

if (dragging)

{

if (bool\_rectangle)

{

if (

((labelForm.Location.X < rectangle.X + rectangle.Width) &&

(labelForm.Location.X > rectangle.X) &&

(labelForm.Location.Y < rectangle.Y + rectangle.Height) &&

(labelForm.Location.Y > rectangle.Y)

) ||

((labelForm.Location.X + labelForm.Width < rectangle.X + rectangle.Width) &&

(labelForm.Location.X + labelForm.Width > rectangle.X) &&

(labelForm.Location.Y < rectangle.Y + rectangle.Height) &&

(labelForm.Location.Y > rectangle.Y)

)

)

bool\_rounded\_rectangle = !bool\_rounded\_rectangle;

bool\_rectangle = false;

}

else if (bool\_circle)

{

if (

((labelForm.Location.X < circle.X + circle.Width) &&

(labelForm.Location.X > circle.X) &&

(labelForm.Location.Y < circle.Y + circle.Height) &&

(labelForm.Location.Y > circle.Y)

) ||

((labelForm.Location.X + labelForm.Width < circle.X + circle.Width) &&

(labelForm.Location.X + labelForm.Width > circle.X) &&

(labelForm.Location.Y < circle.Y + circle.Height) &&

(labelForm.Location.Y > circle.Y)

)

)

bool\_rounded\_circle = !bool\_rounded\_circle;

bool\_circle = false;

}

else

{

if (

((labelForm.Location.X < square.X + square.Width) &&

(labelForm.Location.X > square.X) &&

(labelForm.Location.Y < square.Y + square.Height) &&

(labelForm.Location.Y > square.Y)

) ||

((labelForm.Location.X + labelForm.Width < square.X + square.Width) &&

(labelForm.Location.X + labelForm.Width > square.X) &&

(labelForm.Location.Y < square.Y + square.Height) &&

(labelForm.Location.Y > square.Y)

)

)

bool\_rounded\_square = !bool\_rounded\_square;

bool\_square = false;

}

}

pictureBox.Invalidate();

dragging = false;

}

}

}

**FormMain.Designer.cs**

namespace laboratoryWork\_5

{

partial class FormMain

{

/// <summary>

/// Required designer variable.

/// </summary>

private System.ComponentModel.IContainer components = null;

/// <summary>

/// Clean up any resources being used.

/// </summary>

/// <param name="disposing">true if managed resources should be disposed; otherwise, false.</param>

protected override void Dispose(bool disposing)

{

if (disposing && (components != null))

{

components.Dispose();

}

base.Dispose(disposing);

}

#region Windows Form Designer generated code

/// <summary>

/// Required method for Designer support - do not modify

/// the contents of this method with the code editor.

/// </summary>

private void InitializeComponent()

{

this.pictureBox = new System.Windows.Forms.PictureBox();

this.labelInfo = new System.Windows.Forms.Label();

this.labelForm = new System.Windows.Forms.Label();

this.labelView = new System.Windows.Forms.Label();

((System.ComponentModel.ISupportInitialize) (this.pictureBox)).BeginInit();

this.SuspendLayout();

//

// pictureBox

//

this.pictureBox.Dock = System.Windows.Forms.DockStyle.Fill;

this.pictureBox.Location = new System.Drawing.Point(0, 0);

this.pictureBox.Name = "pictureBox";

this.pictureBox.Size = new System.Drawing.Size(809, 451);

this.pictureBox.TabIndex = 0;

this.pictureBox.TabStop = false;

this.pictureBox.Paint += new System.Windows.Forms.PaintEventHandler(this.pictureBox\_Paint);

this.pictureBox.MouseDown += new System.Windows.Forms.MouseEventHandler(this.pictureBox\_MouseDown);

this.pictureBox.MouseMove += new System.Windows.Forms.MouseEventHandler(this.pictureBox\_MouseMove);

this.pictureBox.MouseUp += new System.Windows.Forms.MouseEventHandler(this.pictureBox\_MouseUp);

//

// labelInfo

//

this.labelInfo.Anchor = ((System.Windows.Forms.AnchorStyles) ((System.Windows.Forms.AnchorStyles.Bottom | System.Windows.Forms.AnchorStyles.Right)));

this.labelInfo.BackColor = System.Drawing.SystemColors.InactiveCaption;

this.labelInfo.FlatStyle = System.Windows.Forms.FlatStyle.Flat;

this.labelInfo.Font = new System.Drawing.Font("Microsoft Sans Serif", 9.75F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte) (204)));

this.labelInfo.Location = new System.Drawing.Point(528, 304);

this.labelInfo.Name = "labelInfo";

this.labelInfo.Padding = new System.Windows.Forms.Padding(30, 20, 30, 20);

this.labelInfo.Size = new System.Drawing.Size(234, 120);

this.labelInfo.TabIndex = 6;

this.labelInfo.Text = "Информация";

this.labelInfo.TextAlign = System.Drawing.ContentAlignment.MiddleCenter;

//

// labelForm

//

this.labelForm.Anchor = System.Windows.Forms.AnchorStyles.Bottom;

this.labelForm.BackColor = System.Drawing.Color.AntiqueWhite;

this.labelForm.BorderStyle = System.Windows.Forms.BorderStyle.Fixed3D;

this.labelForm.FlatStyle = System.Windows.Forms.FlatStyle.Flat;

this.labelForm.Font = new System.Drawing.Font("Microsoft Sans Serif", 9.75F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte) (204)));

this.labelForm.Location = new System.Drawing.Point(312, 332);

this.labelForm.Name = "labelForm";

this.labelForm.Padding = new System.Windows.Forms.Padding(30, 20, 30, 20);

this.labelForm.Size = new System.Drawing.Size(121, 64);

this.labelForm.TabIndex = 5;

this.labelForm.Text = "Форма";

this.labelForm.TextAlign = System.Drawing.ContentAlignment.MiddleCenter;

//

// labelView

//

this.labelView.Anchor = ((System.Windows.Forms.AnchorStyles) ((System.Windows.Forms.AnchorStyles.Bottom | System.Windows.Forms.AnchorStyles.Left)));

this.labelView.BackColor = System.Drawing.Color.AntiqueWhite;

this.labelView.BorderStyle = System.Windows.Forms.BorderStyle.Fixed3D;

this.labelView.FlatStyle = System.Windows.Forms.FlatStyle.Flat;

this.labelView.Font = new System.Drawing.Font("Microsoft Sans Serif", 9.75F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte) (204)));

this.labelView.Location = new System.Drawing.Point(94, 332);

this.labelView.Name = "labelView";

this.labelView.Padding = new System.Windows.Forms.Padding(30, 20, 30, 20);

this.labelView.Size = new System.Drawing.Size(121, 64);

this.labelView.TabIndex = 4;

this.labelView.Text = "Вид";

this.labelView.TextAlign = System.Drawing.ContentAlignment.MiddleCenter;

//

// FormMain

//

this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);

this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;

this.BackColor = System.Drawing.Color.AntiqueWhite;

this.ClientSize = new System.Drawing.Size(809, 451);

this.Controls.Add(this.labelInfo);

this.Controls.Add(this.labelForm);

this.Controls.Add(this.labelView);

this.Controls.Add(this.pictureBox);

this.MinimumSize = new System.Drawing.Size(825, 490);

this.Name = "FormMain";

this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;

this.Text = "Лабораторная работа #5";

((System.ComponentModel.ISupportInitialize) (this.pictureBox)).EndInit();

this.ResumeLayout(false);

}

private System.Windows.Forms.Label labelForm;

private System.Windows.Forms.Label labelInfo;

private System.Windows.Forms.Label labelView;

private System.Windows.Forms.PictureBox pictureBox;

#endregion

}

}