**Лабораторная работа №3**

**«Работа с графикой»**

**Задание 1:**

Разработать программу движения фигуры при помощи таймера. Регулируя интервал работы таймера, можно управлять скоростью движения фигуры. Направление движения фигуры задается с помощью изменения координат по горизонтали и (или) по вертикали.

**Задание 2:**

Учесть, что при изменении размеров формы фигура все равно должна двигаться в заданных направлениях в пределах формы.

Добавьте в проект настройку параметров: выбор цвета фигуры с помощью диалогового окна выбора цвета и настройку скорости движения фигуры. Настройка параметров должна происходить с помощью второй формы.

**Вариант 11.**

Нарисовать пульсирующий круг (увеличивающийся и уменьшающийся), который движется по горизонтали в прямом и обратном направлении. Программа завершается по нажатию любой клавиши.

**Ход выполнения работы**

**Код**

Form1.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 prakt\_lab3

{

public partial class Form1 : Form

{

enum STATUS

{

Left, Right

};

STATUS flag;

Settings settings = new Settings();

Color colorForCircle, colorForForm;

Graphics g;

int x = 100; // локация по x

int w = 100; // размер круна

bool flag2;

public Form1()

{

InitializeComponent();

timer1.Start();

colorForCircle = settings.ColorCircle;

colorForForm = settings.ColorForm;

}

// Метод, срабатывающий при перерисовки элемента управления. Вызывается при событии Paint на Form1

private void Form1\_Paint(object sender, PaintEventArgs e)

{

g = CreateGraphics();

g.Clear(colorForForm);

g.FillEllipse(new SolidBrush(colorForCircle), x, 100, w, w);

}

// Метод, срабатывающий по истечении заданного временного интервала. Вызывается при событии Tick в timer1

private void timer1\_Tick(object sender, EventArgs e)

{

g = CreateGraphics();

g.Clear(colorForForm);

g.FillEllipse(Brushes.White, x, 100, w, w);

this.Invalidate();

if (flag == STATUS.Left) // движение влево

x -= 5;

if (flag == STATUS.Right) // движение вправо

x += 5;

if (x >= (this.ClientSize.Width - w)) // если достигли правого края формы

{

flag = STATUS.Left; // меняем статус движения на левый

}

else if (x <= 1) // если достигли левого края формы

{

flag = STATUS.Right; // меняем статус движения на правый

}

if (flag2 == true)

{

w += 10; // увеличение круга

}

else

{

w -= 10; // уменьшение круга

}

flag2 = !flag2;

}

// Метод, срабатывающий при нажатии любой клавиши на клавиатуре. Вызывается при событии KeyPress на button1

private void button1\_KeyPress(object sender, KeyPressEventArgs e)

{

Close();

}

// Метод, срабатывающий при нажатии левой или правой кнопки мыши. Вызывается при событии Click на Form1

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

{

Close();

}

// Метод, срабатывающий при нажатии на button1. Вызывается при событии Click на button1

private void button1\_Click(object sender, EventArgs e)

{

settings.Owner = this;

settings.ShowDialog();

colorForCircle = settings.ColorCircle; // получение цвета фигуры из настроек

colorForForm = settings.ColorForm; // получение цвета формы из настроек

button1.ForeColor = colorForForm;

timer1.Interval = Convert.ToInt32(150 / settings.SpeedCircle); // получение скорости движения из настроек

}

}

}

Settings.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;

using static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace prakt\_lab3

{

public partial class Settings : Form

{

Color colorForCircle;

Color colorForForm;

int speed;

public Settings()

{

InitializeComponent();

colorForCircle = Color.White;

colorForForm = Color.RoyalBlue;

}

// Цвет фона

public Color ColorForm

{

get

{

return colorForForm;

}

}

// Цвет фигуры

public Color ColorCircle

{

get

{

return colorForCircle;

}

}

// Скорость движения фигуры

public double SpeedCircle

{

get

{

return speed;

}

}

// Метод, срабатывающий при нажатии на button1. Вызывается при событии Click на button1

private void button1\_Click(object sender, EventArgs e)

{

// цвет фигуры

if (comboBox1.SelectedIndex == 0) // красный

{

colorForCircle = Color.Red;

}

else if (comboBox1.SelectedIndex == 1) // зеленый

{

colorForCircle = Color.Green;

}

else if (comboBox1.SelectedIndex == 2) // белый

{

colorForCircle = Color.White;

}

else if (comboBox1.SelectedIndex == 3) // фиолетовый

{

colorForCircle = Color.Purple;

}

else if (comboBox1.SelectedIndex == 4) // серый

{

colorForCircle = Color.Gray;

}

// цвет фона

if (comboBox2.SelectedIndex == 0) // черный

{

colorForForm = Color.Black;

}

else if (comboBox2.SelectedIndex == 1) // синий

{

colorForForm= Color.RoyalBlue;

}

else if (comboBox2.SelectedIndex == 2) // коричневый

{

colorForForm= Color.Brown;

}

else if (comboBox2.SelectedIndex == 3) // оранжевый

{

colorForForm= Color.Orange;

}

else if (comboBox2.SelectedIndex == 4) // розовый

{

colorForForm= Color.Pink;

}

speed = Convert.ToInt32(numericUpDown1.Value);

Close();

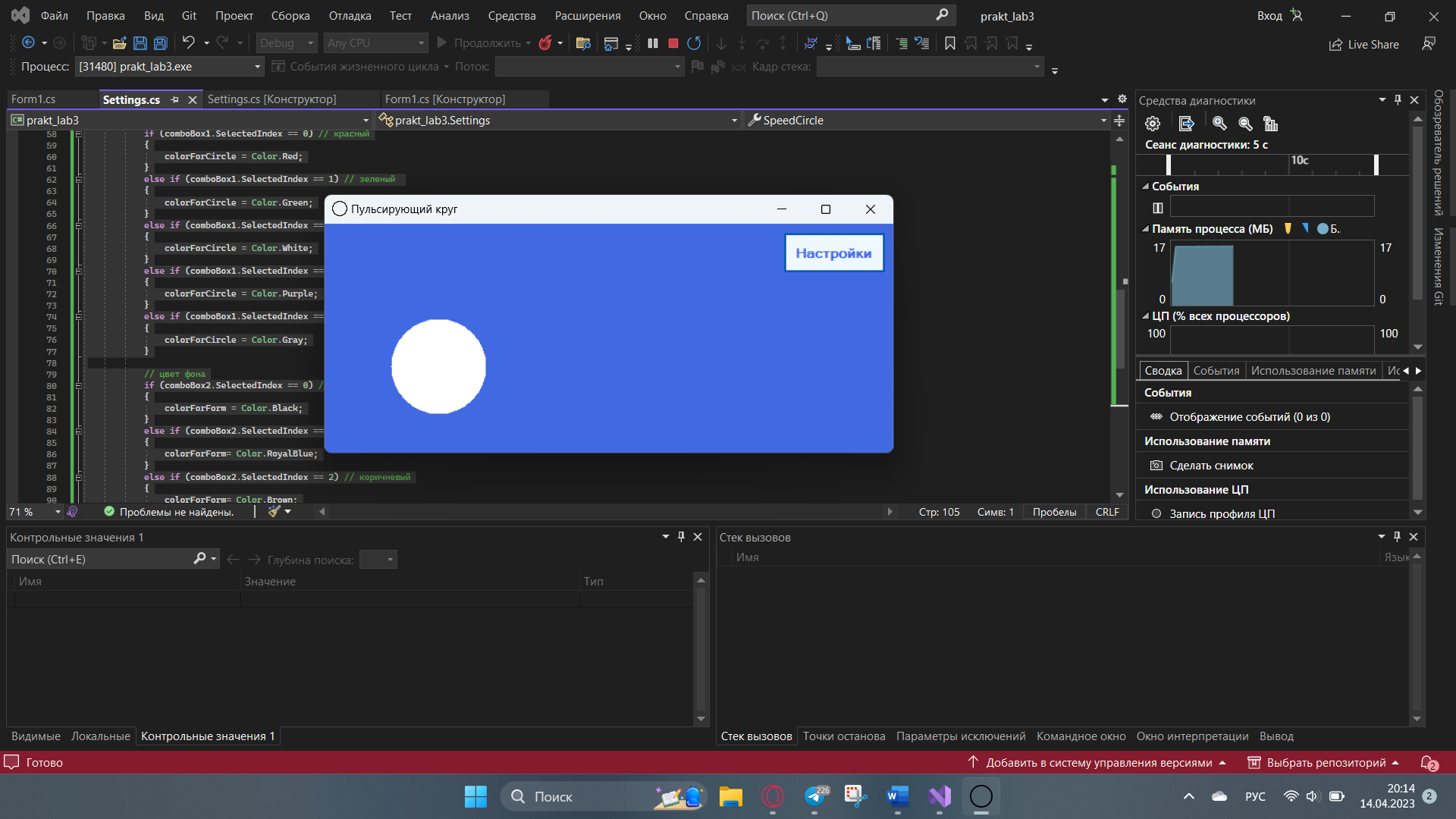
}

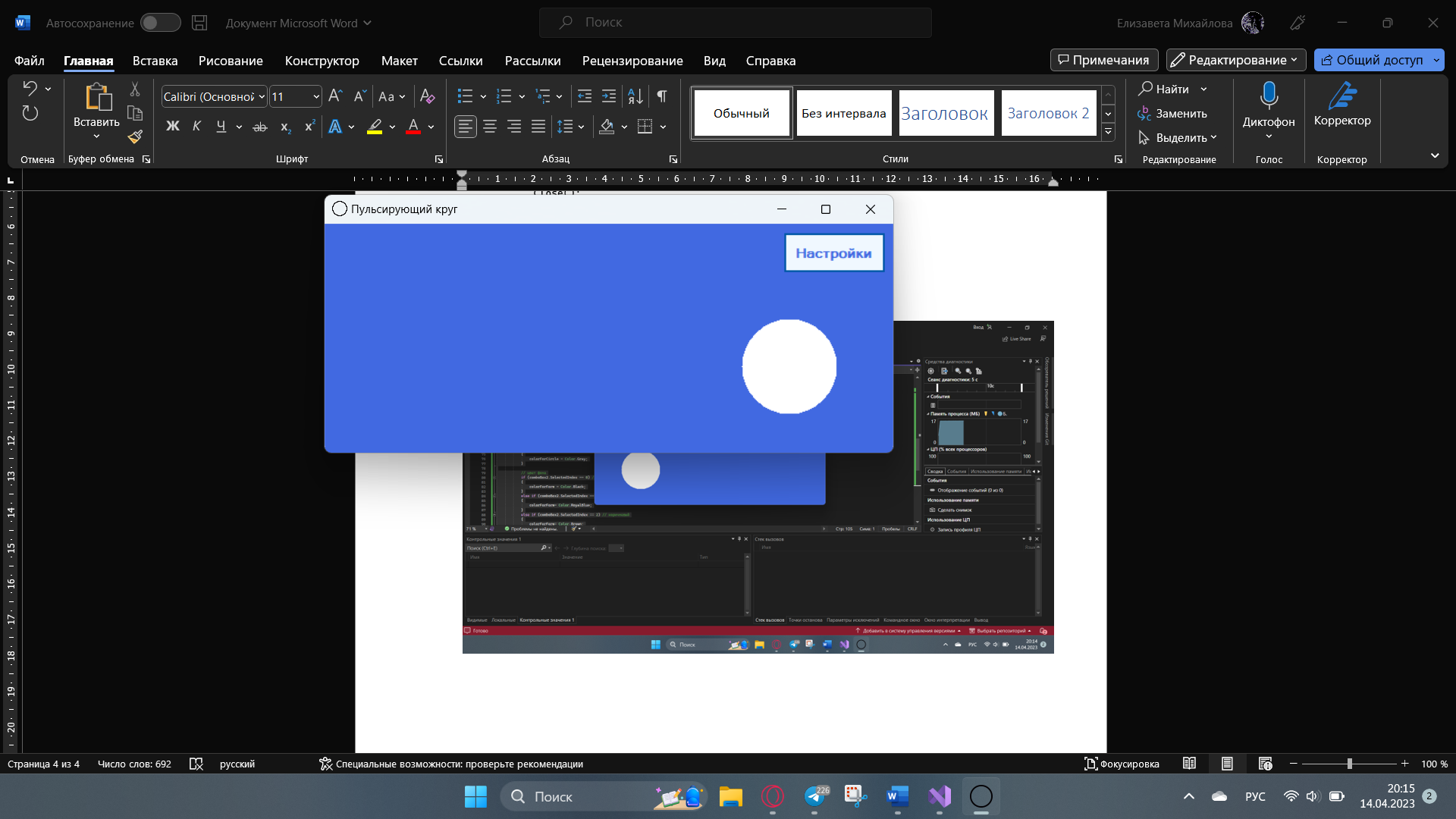
}

}

**Результат**

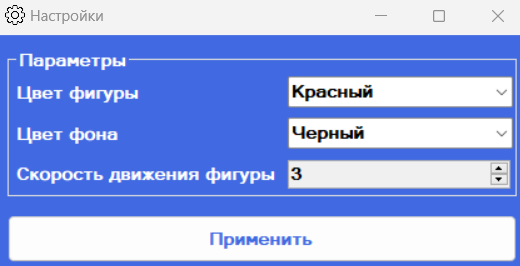
Круг пульсирует и двигается по горизонтали туда обратно







Настройки



Изображение выглядит как текст, снимок экрана, монитор

Автоматически созданное описание