МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «КубГУ»)

Факультет компьютерных технологий и прикладной математики  
Кафедра информационных технологий

Отчёт по лабораторной работе №5

**«Разработка Windows-приложения на языке С#»**

Работу выполнил  
студент 42 группы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Б.М. Ибрагимов

Направление подготовки 01.03.02 Прикладная математика и информатика

Проверил  
канд. техн. наук, доц.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А.Полупанов

Краснодар 2020

**Цель работы:** является изучение и разработка Windows-приложения на языке С#.

**Задача 1:** создать приложение, в котором вводятся координаты окружности и меняются по нажатию кнопки.

**Задача 2:** создать приложение, в котором движется окружность, по нажатию одной кнопки он останавливается, по нажатию второй кнопки считывается значения красной, зеленой и синей компонент из трех Textbox-ов и изменяется цвет окружности.

**Задача 3:** создать приложение, в котором вводятся три переменные x, y, z, а затем по нажатию кнопки рисуется x – кривых, x+y – кривых Безье, x+y+z – окружностей с произвольными координатами.

**Задача 4:** создать приложение, в котором при нажатии одной копки окружность начинает двигать/останавливаться, по нажатию другой - радиус окружности начинает увеличивать (до определенного момента), затем уменьшаться (до определенного момента), после повторного нажатия кнопки изменение радиуса прекращается.

**Задача 5:** создать приложение, в котором окружность движется в том направлении что движется и мышь.

**Задача 6:** создать приложение, в котором окружность начинает двигаться в противоположном направлению движения мыши.

**Задача 7:** создать приложение, в котором направление движения окружности задается нажатием кнопок (стрелок) клавиатуры).

**Задача 8:** создать приложение, в котором по нажатию правой кнопки мыши появляется окружность с произвольными координатами, по нажатию на левую – окружности исчезают по одной.

**ЛИСТИНГ ПРОГРАММЫ**

**Задача 1**

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 lab05 {

public partial class Form1 : Form {

Graphics g;

public Form1() {

InitializeComponent();

}

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

CX.Maximum = pictureBox1.Width;

CY.Maximum = pictureBox1.Height;

R.Maximum = Math.Ceiling(CX.Maximum / 2);

}

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

pictureBox1.Refresh();

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

g.FillEllipse(new SolidBrush(Color.Red), new Rectangle(new Point((int)(CX.Value - R.Value), (int)(CY.Value - R.Value)), new Size((int)(2 \* R.Value), (int)(2 \* R.Value))));

}

}

}

**Задача 2**

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 lab05 {

public partial class Form2 : Form {

Graphics g;

Point locCircle;

Size sizeCircle;

int CR, CG, CB, orientation = 1;

public Form2() {

InitializeComponent();

}

private void StopButton\_Click(object sender, EventArgs e) {

timer1.Enabled = !timer1.Enabled;

}

private void RePaintButton\_Click(object sender, EventArgs e) {

CR = Convert.ToInt32(R.Text);

if (CR > 255) CR = 255;

CG = Convert.ToInt32(G.Text);

if (CG > 255) CG = 255;

CB = Convert.ToInt32(B.Text);

if (CB > 255) CB = 255;

}

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

if (locCircle.X >= pictureBox1.Width - sizeCircle.Width) orientation = -1;

if (locCircle.X <= sizeCircle.Width) orientation = 1;

locCircle = new Point(locCircle.X + orientation, locCircle.Y);

pictureBox1.Refresh();

}

private void Form2\_Load(object sender, EventArgs e) {

sizeCircle = new Size(100, 100);

locCircle = new Point(sizeCircle.Width, (int)(pictureBox1.Height/2));

R.Text = "0"; G.Text = "0"; B.Text = "0";

CR = Convert.ToInt32(R.Text); CG = Convert.ToInt32(G.Text); CB = Convert.ToInt32(B.Text);

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

g.FillEllipse(new SolidBrush(Color.FromArgb(CR,CG,CB)), new Rectangle(locCircle,sizeCircle));

}

}

}

**Задача 3**

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 System.Windows.Forms.DataVisualization.Charting;

namespace lab05 {

public partial class Form3 : Form {

int CX=0, CY=0, CZ=0;

Graphics g;

Pen pen = new Pen(new SolidBrush(Color.Red));

public Form3() {

InitializeComponent();

}

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

chart1.Series.Clear();

CX = Convert.ToInt32(X.Text);

CY = Convert.ToInt32(Y.Text);

CZ = Convert.ToInt32(Z.Text);

DrawCurves(CX);

pictureBox2.Refresh();

pictureBox1.Refresh();

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

Random random = new Random();

Size size = new Size(20, 20);

int countCircles = CX + CY + CZ;

for (int i = 0; i < countCircles; i++) {

Point loc = new Point(random.Next(size.Width, pictureBox1.Width - 2 \* size.Width), random.Next(size.Height, pictureBox1.Height - 2 \* size.Height));

g.FillEllipse(new SolidBrush(Color.Blue), new Rectangle(loc,size));

}

}

private void pictureBox2\_Paint(object sender, PaintEventArgs e) {

int countCurvesBezie = CX + CY;

g = e.Graphics;

for (int i = 0; i < countCurvesBezie; i++) {

int dist = 20 \* i;

Point p1 = new Point(dist, dist);

Point p2 = new Point(pictureBox2.Width - dist, dist);

Point p3 = new Point(dist, pictureBox2.Height - dist);

Point p4 = new Point(pictureBox2.Width - dist, pictureBox2.Height - dist);

g.DrawBezier(new Pen(new SolidBrush(Color.Aqua)),p1,p2,p3,p4);

}

}

private void DrawCurves(int CX) {

double step = 0.01, xmin = -10, xmax = 10;

int count = (int)Math.Ceiling((xmax - xmin) / step);

double[] x = new double[count];

List<Series> y = new List<Series>();

for (int i = 0; i < count; i++) x[i] = xmin + step \* i;

for (int i = 0; i < CX; i++) {

y.Add(new Series());

y[i].ChartType = SeriesChartType.Spline;

for (int j = 0; j < count; j++) {

y[i].Points.AddXY(x[j], Math.Sin((i + 1) \* x[j]));

}

}

chart1.ChartAreas[0].AxisX.Minimum = xmin;

chart1.ChartAreas[0].AxisX.Maximum = xmax;

for (int i = 0; i < CX; i++) chart1.Series.Add(y[i]);

}

}

}

**Задача 4**

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 lab05 {

public partial class Form4 : Form {

Graphics g;

int orientationLoc = 1, orientationSize = 1;

Size size = new Size(50, 50);

Point loc;

public Form4() {

InitializeComponent();

}

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

timer1.Enabled = !timer1.Enabled;

}

private void button2\_Click(object sender, EventArgs e) {

timer2.Enabled = !timer2.Enabled;

}

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

if (loc.X >= pictureBox1.Width - size.Width) orientationLoc = -1;

if (loc.X <= size.Width) orientationLoc = 1;

loc = new Point(loc.X + orientationLoc, loc.Y);

pictureBox1.Refresh();

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

g.FillEllipse(new SolidBrush(Color.BurlyWood), new Rectangle(loc,size));

}

private void Form4\_Load(object sender, EventArgs e) {

loc = new Point(size.Width, pictureBox1.Height / 2);

}

private void timer2\_Tick(object sender, EventArgs e) {

if (size.Width >= 150) orientationSize = -1;

if (size.Width <= 5) orientationSize = 1;

size = new Size(size.Width + orientationSize, size.Height + orientationSize);

pictureBox1.Refresh();

}

}

}

**Задача 5**

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 lab05 {

public partial class Form5 : Form {

Graphics g;

Size size = new Size(50, 50);

Point loc;

public Form5() {

InitializeComponent();

}

private void Form5\_Load(object sender, EventArgs e) {

loc = new Point(size.Width, size.Height);

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

g.FillEllipse(new SolidBrush(Color.Blue), new Rectangle(loc, size));

}

private void pictureBox1\_MouseMove(object sender, MouseEventArgs e) {

loc = e.Location;

pictureBox1.Refresh();

}

}

}

**Задача 6**

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 lab05 {

public partial class Form6 : Form {

Graphics g;

double x2, y2;

double orientX = 0, orientY = 0;

Size size = new Size(50, 50);

Point loc;

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

g.FillEllipse(new SolidBrush(Color.BurlyWood), new Rectangle(loc, size));

}

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

Point newLoc = new Point((int)(loc.X - orientX), (int)(loc.Y - orientY));

if (newLoc.X >= pictureBox1.Width - 2 \* size.Width)

newLoc = new Point(pictureBox1.Width - 2 \* size.Width, newLoc.Y);

if (newLoc.X <= 2 \* size.Width)

newLoc = new Point(2 \* size.Width, newLoc.Y);

if (newLoc.Y >= pictureBox1.Height - 2 \* size.Height)

newLoc = new Point(newLoc.X, pictureBox1.Height - 2 \* size.Height);

if (newLoc.Y <= 2 \* size.Height)

newLoc = new Point(newLoc.X, 2 \* size.Height);

loc = newLoc;

pictureBox1.Refresh();

orientX = 0;

orientY = 0;

}

private void Form6\_Load(object sender, EventArgs e) {

loc = new Point(pictureBox1.Width / 2, pictureBox1.Height / 2);

x2 = loc.X;

y2 = loc.Y;

}

public Form6() {

InitializeComponent();

}

private void pictureBox1\_MouseMove(object sender, MouseEventArgs e) {

orientX = e.X - x2;

orientY = e.Y - y2;

x2 = e.X;

y2 = e.Y;

}

}

}

**Задача 7**

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 lab05 {

public partial class Form7 : Form {

Graphics g;

Size size = new Size(50, 50);

Point loc;

double step = 1;

public Form7() {

InitializeComponent();

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

g.FillEllipse(new SolidBrush(Color.BlueViolet), new Rectangle(loc, size));

}

private void Form7\_Load(object sender, EventArgs e) {

loc = new Point(pictureBox1.Width / 2, pictureBox1.Height / 2);

}

private void Form7\_KeyDown(object sender, KeyEventArgs e) {

if (e.KeyCode == Keys.Left) loc = new Point((int)(loc.X - step), loc.Y);

if (e.KeyCode == Keys.Right) loc = new Point((int)(loc.X + step), loc.Y);

if (e.KeyCode == Keys.Down) loc = new Point(loc.X, (int)(loc.Y + step));

if (e.KeyCode == Keys.Up) loc = new Point(loc.X, (int)(loc.Y - step));

pictureBox1.Refresh();

}

}

}

**Задача 8**

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 lab05 {

public partial class Form8 : Form {

Graphics g;

Size size = new Size(50, 50);

List<Point> locs = new List<Point>();

Random random = new Random();

public Form8() {

InitializeComponent();

}

private void pictureBox1\_Paint(object sender, PaintEventArgs e) {

g = e.Graphics;

foreach (Point p in locs) {

g.FillEllipse(new SolidBrush(Color.Aquamarine), new Rectangle(p, size));

}

}

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

if (e.Button == MouseButtons.Right) {

locs.Add(new Point(random.Next(size.Width, pictureBox1.Width - size.Width), random.Next(size.Height, pictureBox1.Height - size.Height)));

pictureBox1.Refresh();

}

if (e.Button == MouseButtons.Left)

if (locs.Count > 0) {

locs.RemoveAt(locs.Count - 1);

pictureBox1.Refresh();

}

else

MessageBox.Show("Все окружности уже удалены");

}

}

}

**РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ**

**Задача 1**

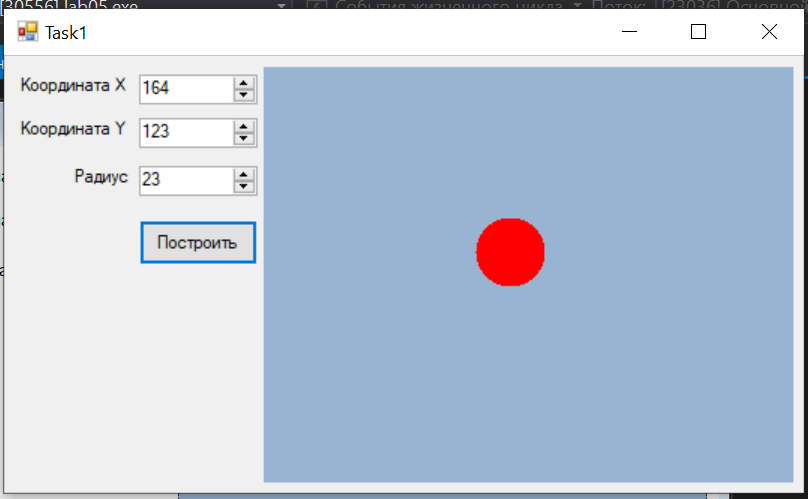
**

Рисунок 1 – результат выполнения задачи 1

**Задача 2**

*Изображение выглядит как снимок экрана

Автоматически созданное описание*

Рисунок 2 – результат выполнения задачи 2

**Задача 3**

*Изображение выглядит как снимок экрана, цветной

Автоматически созданное описание*

Рисунок 3 – результат выполнения задачи 3

**Задача 4**

*Изображение выглядит как снимок экрана

Автоматически созданное описание*

Рисунок 4 – результат выполнения задачи 4

**Задача 5**

*Изображение выглядит как снимок экрана

Автоматически созданное описание*

Рисунок 5 – результат выполнения задачи 5

**Задача 6**

*Изображение выглядит как снимок экрана

Автоматически созданное описание*

Рисунок 6 – результат выполнения задачи 6

**Задача 7**

*Изображение выглядит как снимок экрана

Автоматически созданное описание*

Рисунок 7 – результат выполнения задачи 7

**Задача 8**

*Изображение выглядит как снимок экрана

Автоматически созданное описание*

Рисунок 8 – результат выполнения задачи 8

**Вывод:** язык С# позволяет разрабатывать Windows-приложения с возможностью добавления в форму различных компонент и отслеживания разных действий, включая нажатие кнопки, движение мыши, рисование, работа по таймеру и т. д.