Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«Уфимский университет науки и технологий»

Факультет информатики и робототехники

Кафедра Информатики

Отчёт по лабораторной работе №4.1

По дисциплине «Объектно-ориентированное программирование»

Выполнил:

Студент группы ПРО-231 К.И. Арменшин

Проверил:

доцент кафедры ВМиК Котельников В.А.

Уфа – 2023

**Цель**

Создать приложение для рисования и удаление кругов на форме.

**Задание**

* Создать простейшее приложение с GUI, содержащее:
  + определение простейшего класса CCircle с координатами и постоянным радиусом;
  + форму с объектом для рисования (например, PaintBox);
  + два элемента checkbxox для управления поведением
* Реализовать следующее поведение:
  + при нажатии мышкой на форме создается новый объект CCircle с координатами нажатия и помещается в контейнер;
  + при событии Paint должны отрисовываться на форме все объекты из контейнера
* Реализовать следующее поведение с выделением объектов:
  + хотя бы один объект из существующих на форме всегда является выделенным и отрисовывается отлично от других объектов;
  + выделенными могут быть несколько объектов;
  + выделение происходит при нажатии ЛКМ (левой клавиши мыши) на объект CCircle на форме; при нажатии на область формы, где пересекаются несколько кругов, могут выделяться они все или какой-то один из них
  + при нажатии на кнопку Del, все выделенные объекты должны удаляться
  + при выделении объекта с помощью ЛКМ и удерживаемой клавиши Ctrl, выделенными становятся несколько объектов
  + несколько изменяемых пользователем флагов (элементы checkbox) управляют логикой поведения:
    - checkbox для указания, работает ли клавиша Ctrl
    - checkbox для указания, выделяется только один объект при нажатии на их пересечение, или все, в которые попала мышка.

**Ход работы**

Создание кругов по нажатию мыши. Самый последний созданный элемент становится выделенным.

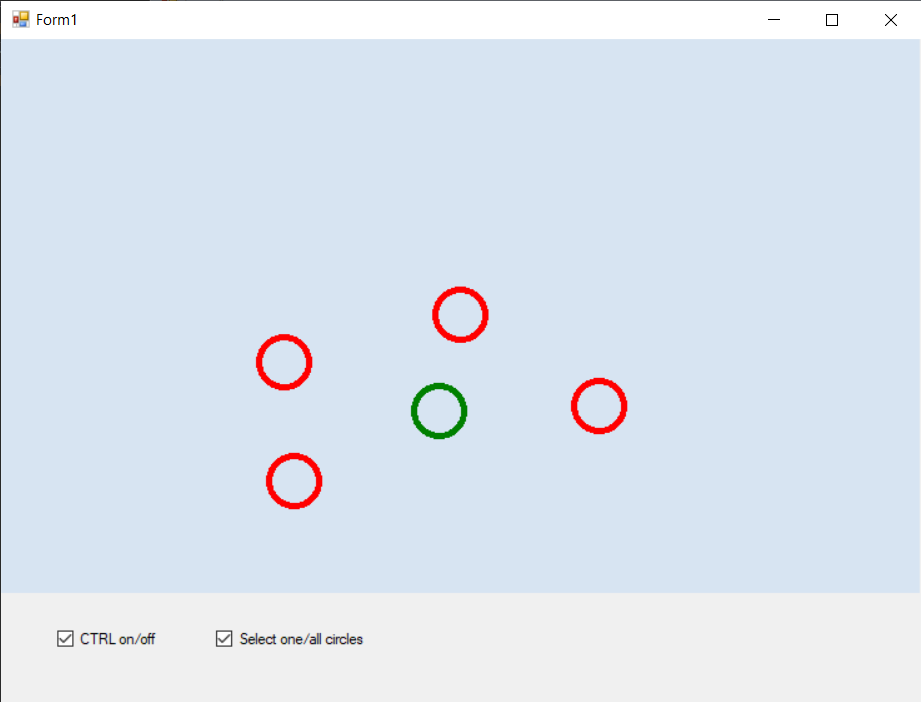


Рисунок . Создание кругов

Выделение кругов при зажатой клавише CTRL и включённым чекбоксом “CTRL on/off”.

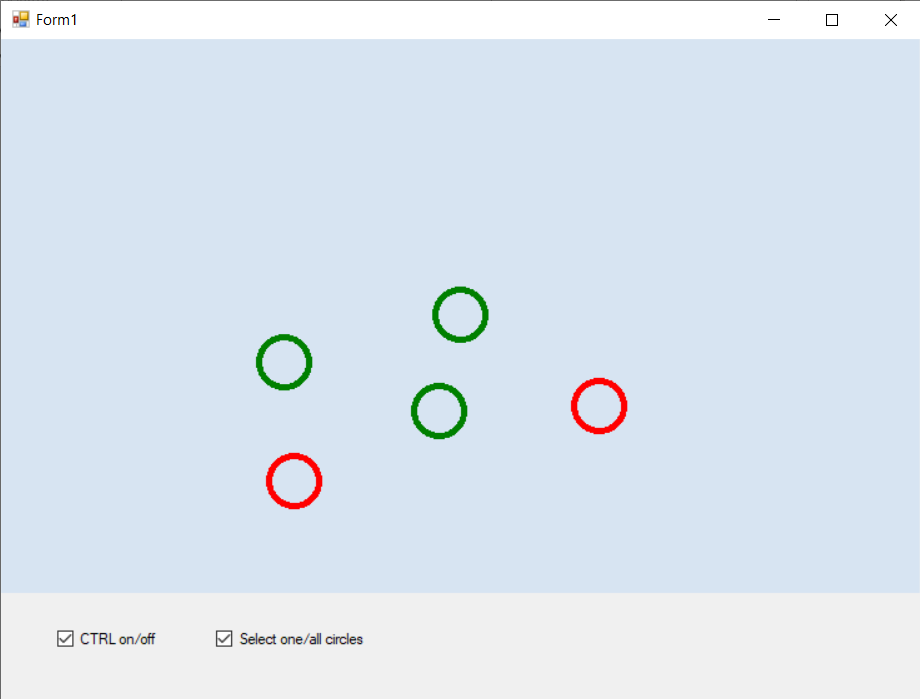


Рисунок . Выделение кругов с помощью CTRL

Выделение нескольких пересекающихся кругов при включённом чекбоксе “Select one/all circles”.

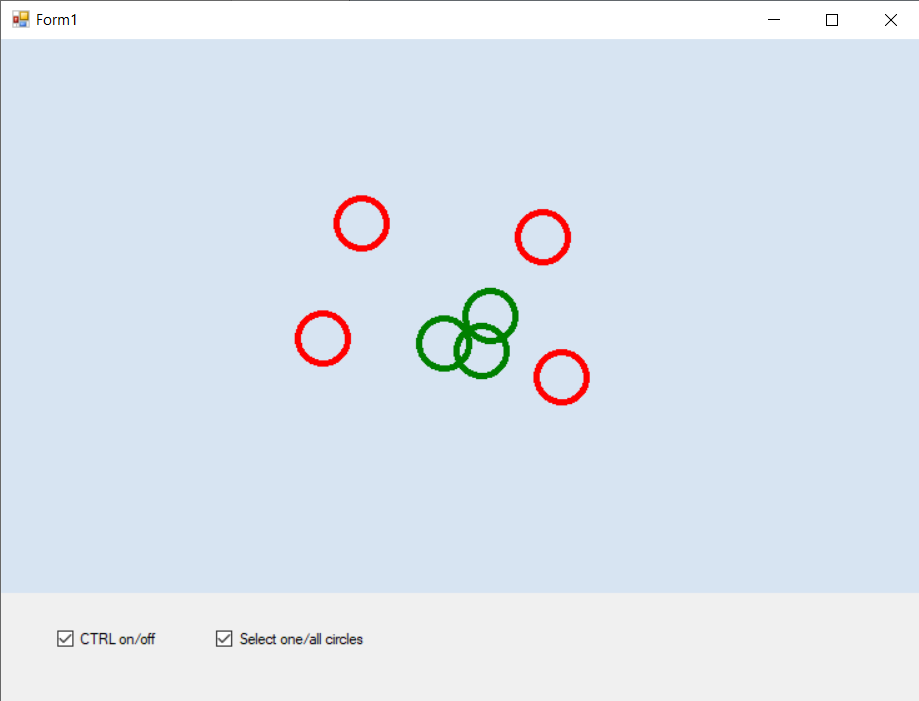


Рисунок . Выделение пересекающихся кругов

Удаление выделенных кругов.

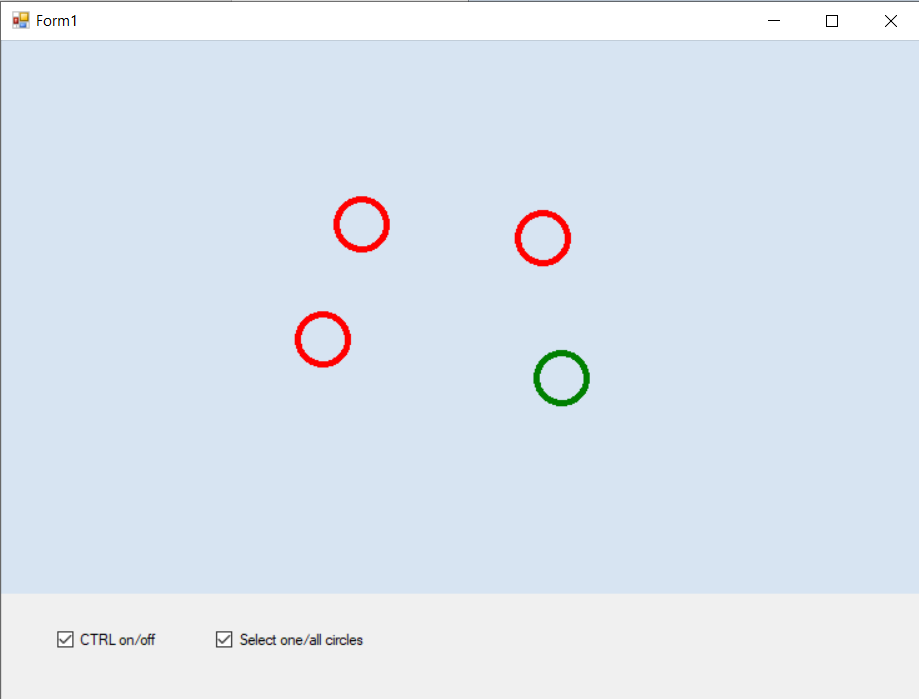


Рисунок . Удаление кругов

**Вывод**

В результате выполнения данной лабораторной работы было разработано приложение, рисующее и удаляющее круги на форме.

Полный код находится в Github:

https://github.com/karimarm/Labs\_OOP.git

**Приложение 1**

L4\_ECG.cpp

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 L4\_OOP

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

penCircleMake();

}

static Pen penCircleSelect = new Pen(Brushes.Green);

static Pen penCircleNotSelect = new Pen(Brushes.Red);

void penCircleMake()

{

penCircleSelect.Width = 5;

penCircleNotSelect.Width = 5;

}

static int radius = 20;

bool isCtrl = false;

class CCircle // Класс кругов

{

public Rectangle rect;

private bool isSelect;

public CCircle(int x, int y) // Конструктор

{

rect = new Rectangle(x, y, radius \* 2, radius \* 2);

isSelect = true;

}

public void draw(PaintEventArgs e) // Отрисовка кругов

{

Graphics graphic = e.Graphics;

graphic.DrawEllipse((isSelect ? penCircleSelect : penCircleNotSelect), rect);

}

public bool isClicked(int x, int y) // Проверка нажатия на круг

{

if (Math.Abs(rect.X - x) <= radius && Math.Abs(rect.Y - y) <= radius)

return true;

else

return false;

}

public void setSelect(bool select) // Установка флага выбранности

{

isSelect = select;

}

public void changeSelect() // Изменение флага выбранности

{

isSelect = !isSelect;

}

public bool getSelect()

{

return isSelect;

}

}

List<CCircle> Circles = new List<CCircle>(); // Список кругов

private void pbCircles\_MouseClick(object sender, MouseEventArgs e)

{

bool flCircleClick = false;

if (!isCtrl || !cbIsCtrl.Checked)

foreach (var j in Circles)

j.setSelect(false);

foreach (var i in Circles) // Проверка кругов на нажатие

{

if (i.isClicked(e.X - radius, e.Y - radius))

{

i.changeSelect();

flCircleClick = true;

if(!cbSelectionCircles.Checked)

break;

}

}

if (!flCircleClick) // Создание нового круга

{

if (!isCtrl || !cbIsCtrl.Checked)

foreach (var j in Circles)

j.setSelect(false);

Circles.Add(new CCircle(e.X - radius, e.Y - radius));

}

pbCircles.Invalidate();

}

private void pbCircles\_Paint(object sender, PaintEventArgs e) // Событие отрисовки

{

foreach (var i in Circles)

i.draw(e);

}

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Delete) // Удаление кругов

{

for (int i = 0; i<Circles.Count(); ++i)

{

if (Circles[i].getSelect())

{

Circles.RemoveAt(i--);

}

}

if(Circles.Count != 0)

Circles.Last().setSelect(true);

pbCircles.Invalidate();

}

else if (e.Control) // Нажатие CTRL

{

isCtrl = true;

}

}

private void Form1\_KeyUp(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.ControlKey) // Отпускание CTRL

{

isCtrl = false;

}

}

}

}