Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский государственный технический университет»  
Кафедра ИИТ

Лабораторная работа №2

“Проектирование и реализация многооконных приложений на языке С#”  
По дисциплине: «**Объектное моделирование интеллектуальных систем** »

Выполнил:  
студент 3 курса  
группы ИИ-21(I)  
Парфеевец И.А.

Проверила:  
Демидович А.Г.

Брест 2023

ЦЕЛЬ РАБОТЫ: Разработка оконных приложений, использование элементов управления (ЭУ)

**Задание 10.** *Создайте каркас приложения (по указанию преподавателя повторите предыдущие пункты 7-9). Добавьте обработчики сообщений активизации Activated и деактивизации Deactivate формы с выводом в КО значений двух переменных A и DA в соответствии с образцом на рисунке 6.*

Код программы :

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

MessageBox.Show("Форма заугржена!");

}

private void Form1\_FormClosed(object sender, FormClosedEventArgs e)

{

MessageBox.Show("Форма закрыта!");

}

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

{

MessageBox.Show("X:"+e.X+"Y:"+e.Y);

this.Refresh();

}

private void Form1\_KeyPress(object sender, KeyPressEventArgs e)

{

MessageBox.Show("Нажата клавиша: " + e.KeyChar.ToString());

if (e.KeyChar == (char)27)

{

Application.Exit();

}

}

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

{

MessageBox.Show("Форма показана");

}

private int drawAmount = 0;

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

{

Graphics graphics= e.Graphics;

drawAmount++;

Font font = new Font("AriaL",10);

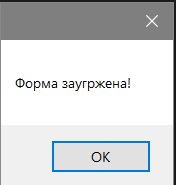
SolidBrush myBrush = new SolidBrush(Color.Black);

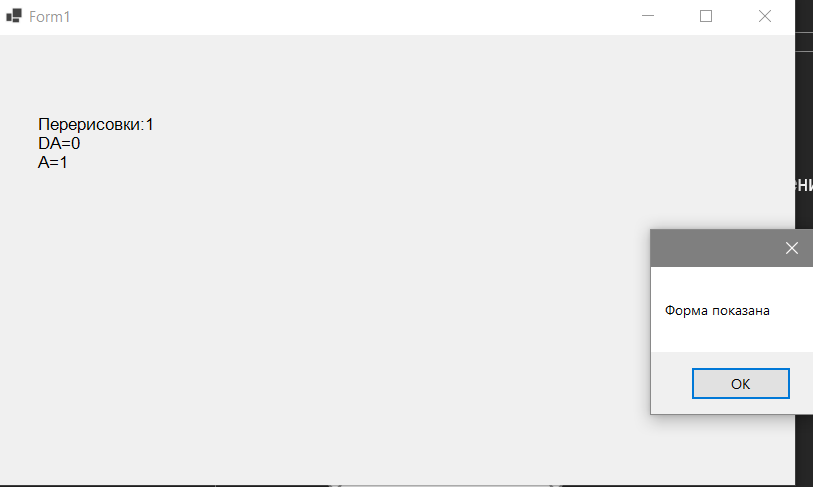
Point point = new Point(40, 80);

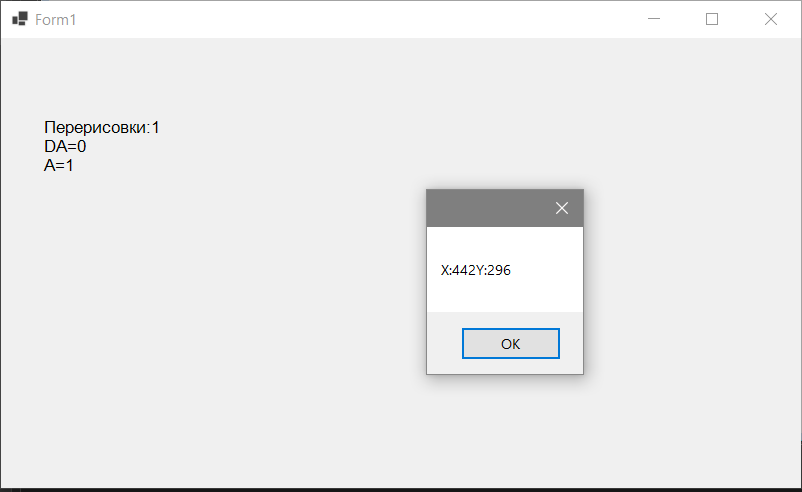
graphics.DrawString("Перерисовки:"+drawAmount.ToString()+"\n"+"DA=0"+"\n"+"A=1", font, myBrush, point);

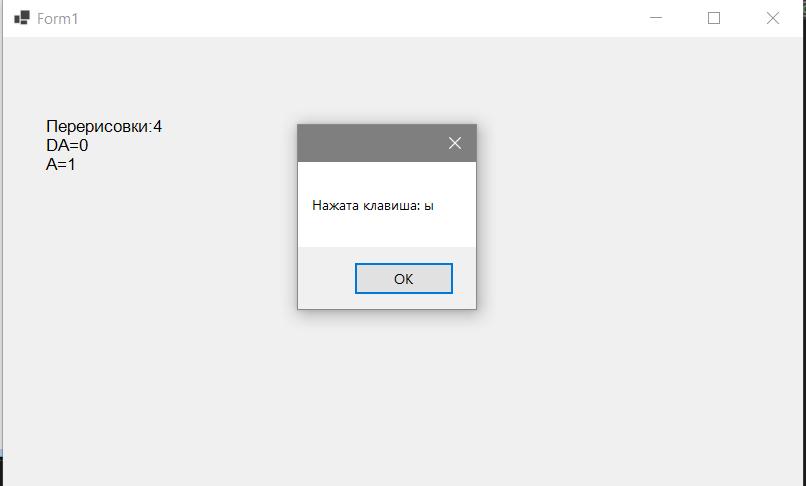
}

Результат работы программы:









**Задание 13.** *Разработайте приложение для работы с числами и строками с интерфейсом, приведенным на рисунке 7*

Код программы :

public Form1()

{

InitializeComponent();

}

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

{

try

{

int value = Convert.ToInt32(valueTB.Text);

double result = Convert.ToDouble( Math.Sqrt(value));

resultTB.Text = result.ToString();

}

catch (Exception ex )

{

MessageBox.Show(ex.ToString());

return;

}

}

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

{

if (listBox1.SelectedIndex != -1){

listBox1.Items.RemoveAt(listBox1.SelectedIndex);

}

}

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

{

if(toAddTextBox.Text.Length > 0)

{

listBox1.Items.Add(toAddTextBox.Text);

toAddTextBox.Text = "";

}

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

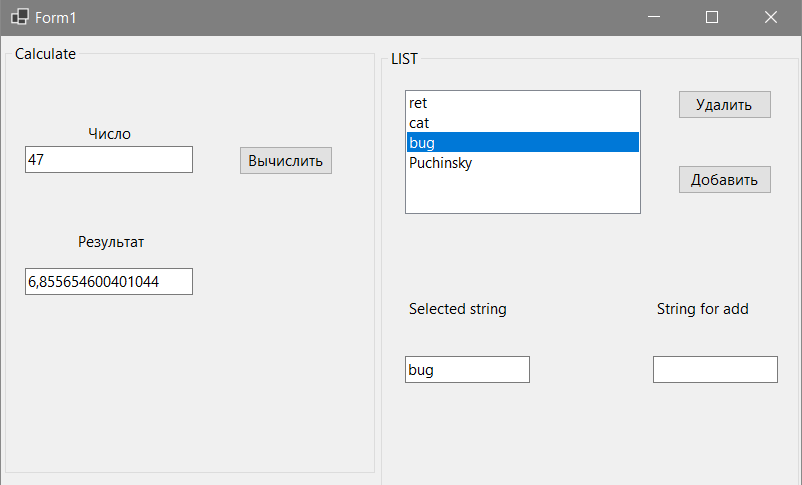
if (listBox1.SelectedIndex != -1)

{

selectedTB.Text = listBox1.Items[listBox1.SelectedIndex].ToString();

}

Результат работы программы:



**Задание 17.** *Создайте аналогичное Заданию 16 приложение с меню (рисунок 13), выполняющее роль упрощенного текстового редактора на базе ЭУ RichTextBox, расположенного поверх КО главной формы.*

Данное приложение было разработано и представлено в отчете по лабораторной работе №1,вместе с заданием №16

**Задание 20.** *Создайте приложение с формой в качестве главного окна и двумя диалоговыми окнами, загружаемыми по щелчкам левой и правой клавиш мыши. Диалоговые окна интегрировать в классе основной формы: для первого ДО использовать зависимость типа использования, а для второго ДО - агрегацию.*

Код программы :

namespace WinFormsApp7

{

public partial class Form1 : Form

{

private DependencyDialog dependencyDialog= new DependencyDialog();

private AggregationDialog aggregationDialog = new AggregationDialog();

public Form1()

{

InitializeComponent();

}

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

{

if (e.Button == MouseButtons.Left)

{

dependencyDialog.SetMessage("Открыто через зависимость типа использования.");

dependencyDialog.ShowDialog();

}

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

{

aggregationDialog.SetMessage("Открыто через агрегацию.");

aggregationDialog.ShowDialog();

}

}

}

}

Код ДОкон:

namespace WinFormsApp7

{

public partial class AggregationDialog : Form

{

public AggregationDialog()

{

InitializeComponent();

}

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

{

}

public void SetMessage(string message)

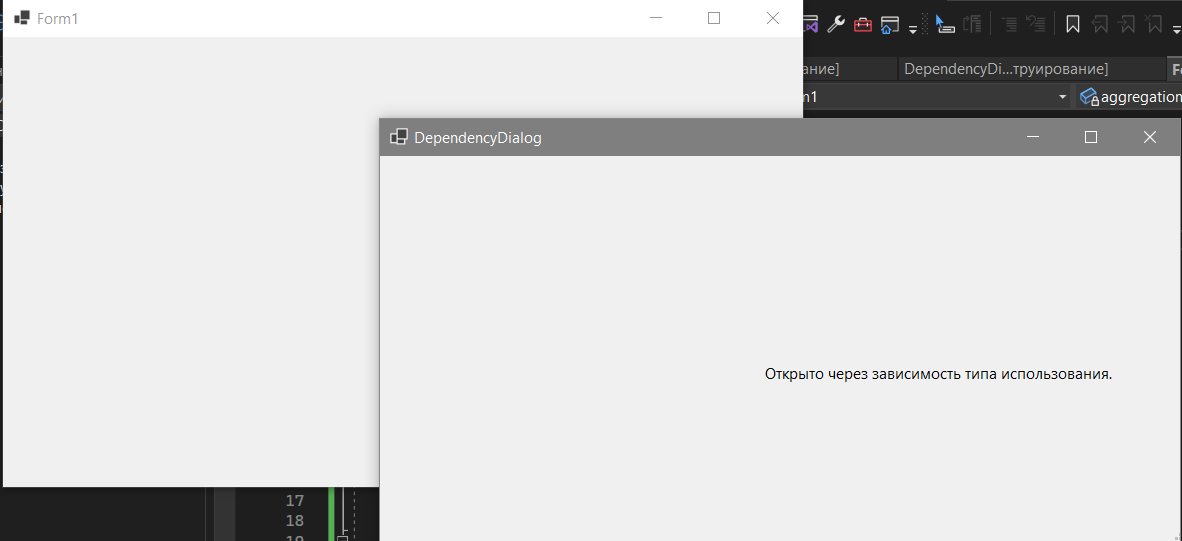
{

label1.Text = message;

}

}

}



Вывод :Разработал оконные приложения, используя элементы управления (ЭУ)