

Отчет по Практической № 3

Тестирование ИС



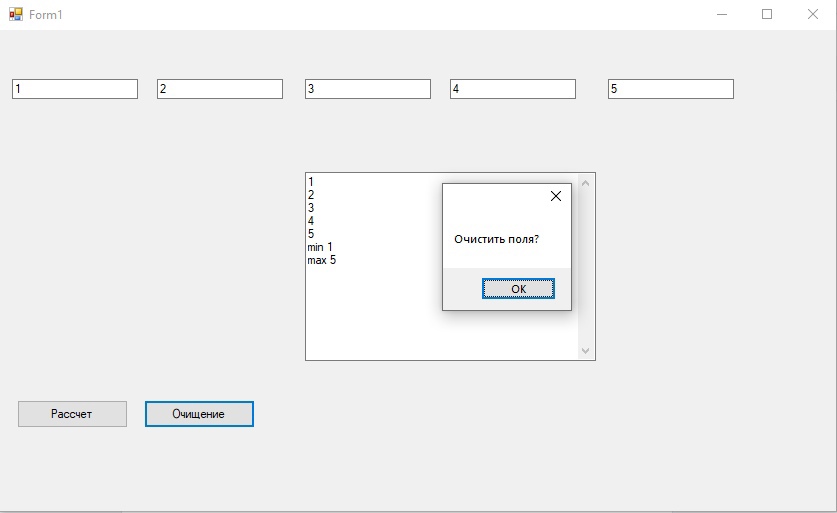
27 ноября 2023 г.

Вадова М.Д

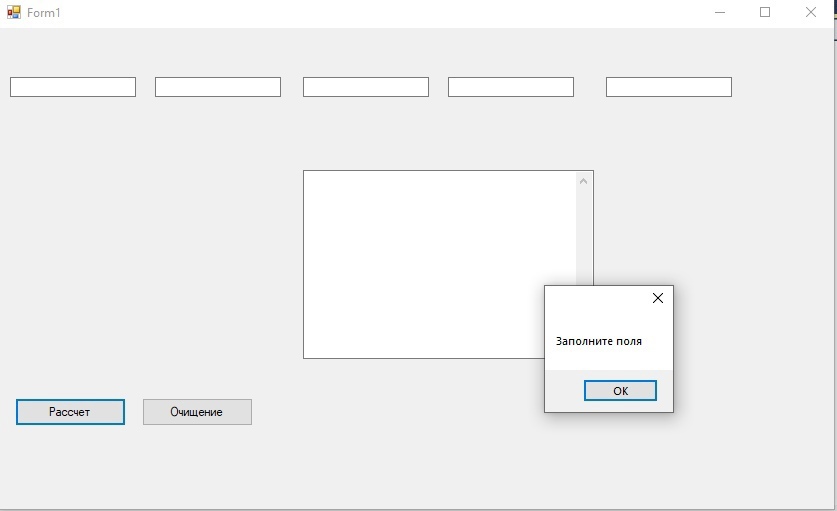
Исправление найденных ошибок из «Практической №2»

1. При нажатии кнопки выскакивает поле "Очистить поля?" При нажатии "Ок" повторно выполняет цикл.

1.1. Добавили еще 1 кнопку на форму для очищения полей, задали ей задачу очищать поле если в нем что то есть.

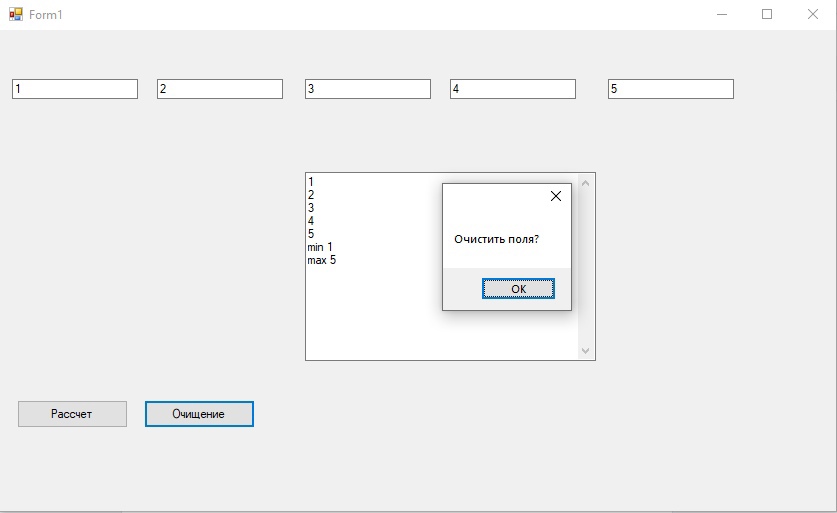


2. При нажатии «Расчет» ничего не вылазило(если в полях ничего нет)

2.1. При нажатии «Расчет» если в поле ничего нет, появляется MessageBox «Заполните поля» 

3. При нажатии кнопки «Очищение» очищался только TextBox6(квадрат в центре), а не все TextBox’ы.

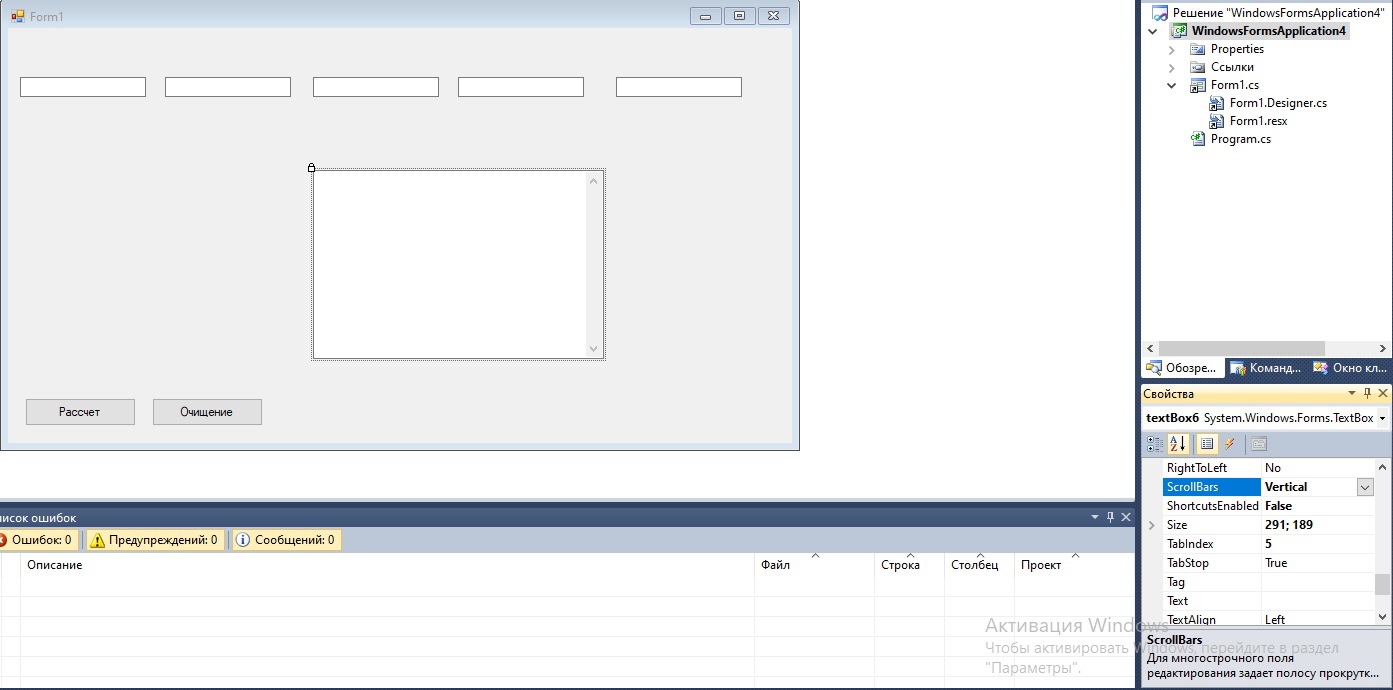
3.1. Создали массив TextBox’ов для очистки всех полей, а не только 6го.

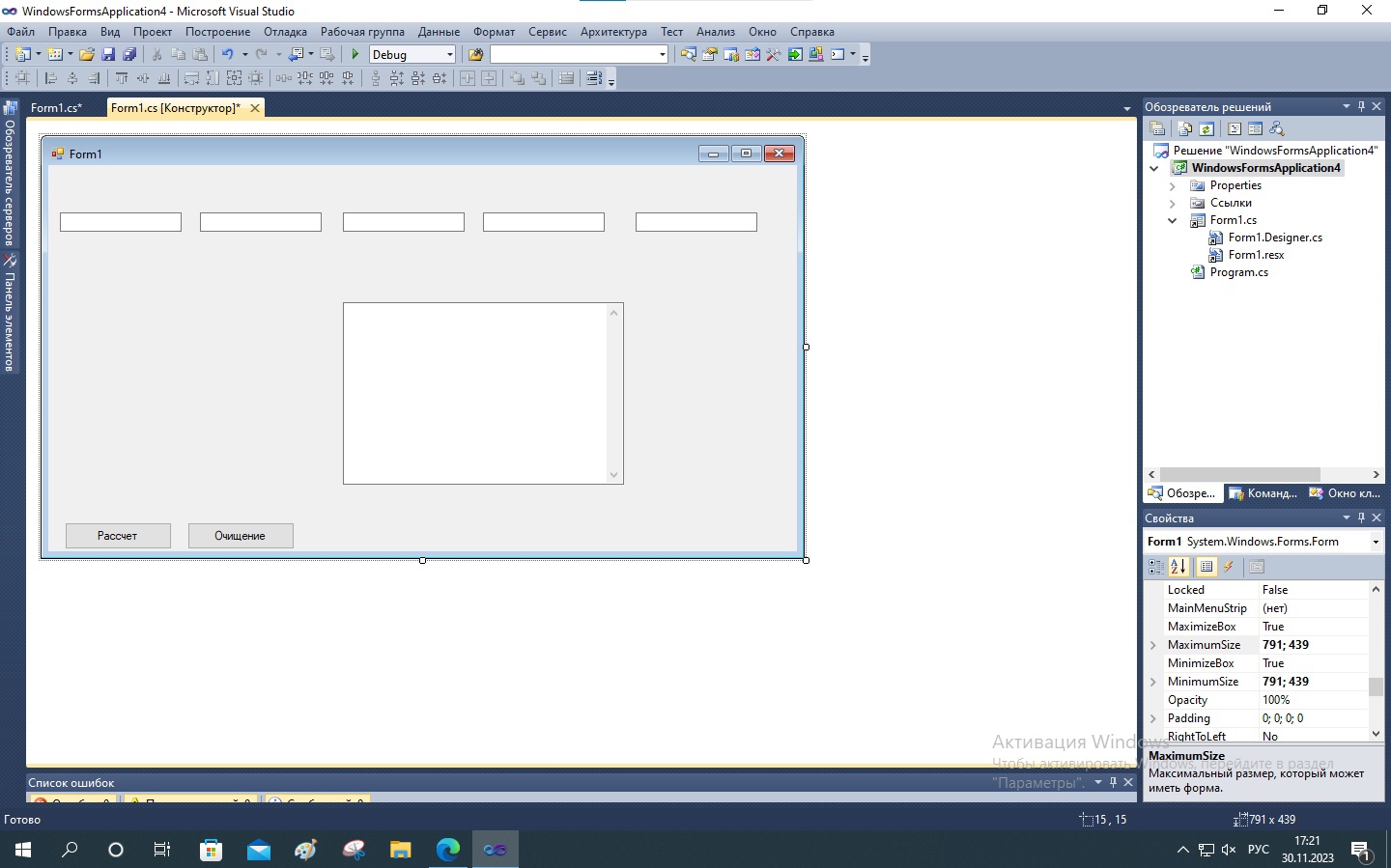


Дополнительно:

Добавили в свойствах ползунок на textbox6 и якорь чтобы не увеличивались textBox'ы.

Добавили в свойствах MinimalSize и MaximumSize для формы чтобы не уменьшалась и не увеличивалась.





***КОД:***

// Начало определения пространств имен, используемых в проекте

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

// Определение пространства имен проекта

namespace WindowsFormsApplication4

{

// Определение класса формы

public partial class Form1 : Form

{

// Определение массива и текстовых полей

int[] array = new int[5];

TextBox[] arrayBox = new TextBox[6];

// Конструктор класса формы

public Form1()

{

// Инициализация компонентов формы

InitializeComponent();

}

// Определение функции сортировки пузырьком

void BubbleSort()

{

int buf;

// Цикл, выполняющий проход по массиву

for (int i = 0; i < array.Length; i++)

{

// Цикл, выполняющий сравнение и перестановку элементов массива

for (int j = i + 1; j < array.Length; j++)

{

if (array[i] > array[j])

{

buf = array[i];

array[i] = array[j];

array[j] = buf;

}

}

}

}

// Обработчик события загрузки формы

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

{

}

// Обработчик события нажатия на кнопку

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

{

// Проверка, является ли введенное значение числом

bool z = int.TryParse(textBox1.Text, out array[0]);

if (z == false)

MessageBox.Show("Заполните поля");

try

{

// Конвертация строковых значений текстовых полей в числа и заполнение массива

array[0] = Convert.ToInt32(textBox1.Text);

array[1] = Convert.ToInt32(textBox2.Text);

array[2] = Convert.ToInt32(textBox3.Text);

array[3] = Convert.ToInt32(textBox4.Text);

array[4] = Convert.ToInt32(textBox5.Text);

// Вызов функции сортировки

BubbleSort();

// Присвоение текстовым полям названий массива

arrayBox[0] = textBox1;

arrayBox[1] = textBox2;

arrayBox[2] = textBox3;

arrayBox[3] = textBox4;

arrayBox[4] = textBox5;

arrayBox[5] = textBox6;

// Проверка, не пустое ли поле textbox6

if (textBox6.Text != "")

{

MessageBox.Show("");

}

// Вывод элементов массива в текстовое поле

for (int i = 0; i < array.Length; i++)

{

textBox6.Text += array[i].ToString() + "\r\n";

}

textBox6.Text += "min " + array[0].ToString() + "\r\n";

textBox6.Text += "max " + array[4].ToString() + "\r\n";

{

}

}

// Обработка исключений

catch (Exception ex)

{

MessageBox.Show(ex.Message);

//throw;

}

}

// Обработчик события изменения текстового поля

private void textBox6\_TextChanged(object sender, EventArgs e)

{

//if (textBox6.Text != null && )

//{

// textBox6.Clear();

// else

//}

}

// Обработчик события изменения текстового поля

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

// Обработчик события нажатия на кнопку

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

{

private void textBox6\_TextChanged(object sender, EventArgs e)

{

// Обработчик события изменения текста в текстовом поле textBox6

// При каждом изменении текста в поле textBox6 будет вызываться этот метод

// Код внутри данного метода будет выполняться при каждом изменении текста

// в поле textBox6

//if (textBox6.Text != null && )

//{

// Проверка, что текст в поле textBox6 не является пустым или null

// Если текст не является пустым или null, то выполняется код внутри блока if

//textBox6.Clear();

// Очищение текста в поле textBox6

// else

//}

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

// Обработчик события изменения текста в текстовом поле textBox1

}

//

// Обработчик события изменения текста в textBox6

//

private void textBox6\_TextChanged(object sender, EventArgs e)

{

// Если текст в textBox6 не равен null и ...

//{

// Очищаем содержимое textBox6

// else

//}

}

//

// Обработчик события изменения текста в textBox1

//

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

// Обработчик нажатия на кнопку button2

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

{

// Выводим диалоговое окно с вопросом "Очистить поля?"

DialogResult result = MessageBox.Show("Очистить поля?");

// Если пользователь нажал "ОК"

if (result == DialogResult.OK)

{

// Перебираем все элементы массива arrayBox

for (int m = 0; m < arrayBox.Length; m++)

{

// Если текст в текущем элементе массива не пустой

if (arrayBox[m].Text != "")

{

// Очищаем содержимое текущего элемента массива

arrayBox[m].Clear();

}

else

{

}

}

}

}