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 №4.2

По дисциплине «Объектно-ориентированное программирование»

Выполнил:

Студент группы ПРО-231 К.И. Арменшин

Проверил:

доцент кафедры ВМиК Котельников В.А.

Уфа – 2023

**Цель**

Создать MVC приложение.

**Задание**

* Создать простейшее приложение с GUI, содержащее:
  + три целых числа A, B и C со значением в пределах от 0 до 100;
  + каждое из чисел должно отображаться и редактироваться в 3 разных компонентах: в textBox, numericUpDown, trackBar (или аналогичных в других языках), при этом редактирование числа в одном поле должно приводить к изменению отображения этого числа во всех других полях;
  + второе число всегда должно быть не меньше первого и не больше третьего;
  + приложение должно сохранять значения чисел между запусками (запоминать при закрытии и восстанавливать значения при открытии);
* Разработанное приложение должно быть реализовано в стиле MVC:
  + хранение трёх чисел должно быть организовано в виде отдельного объекта-модели;
  + все пересчёты, проверки и сохранение должны выполняться в объекте-модели;
  + изменение A и C должно реализовывать разрешающее поведение (при нарушении ограничений порядка модель сама перестраивается, чтобы их выполнить, введённое пользователем значение A и C сохраняется)
  + изменение B должно реализовывать запрещающее поведение (при нарушении ограничений порядка модель откатывает внесённые пользователем изменения) или ограничивающее поведение (при нарушении ограничений порядка модель корректирует введённое значение так, чтобы оно было как можно ближе к заданному пользователем).

**Ход работы**

Внешний вид формы.

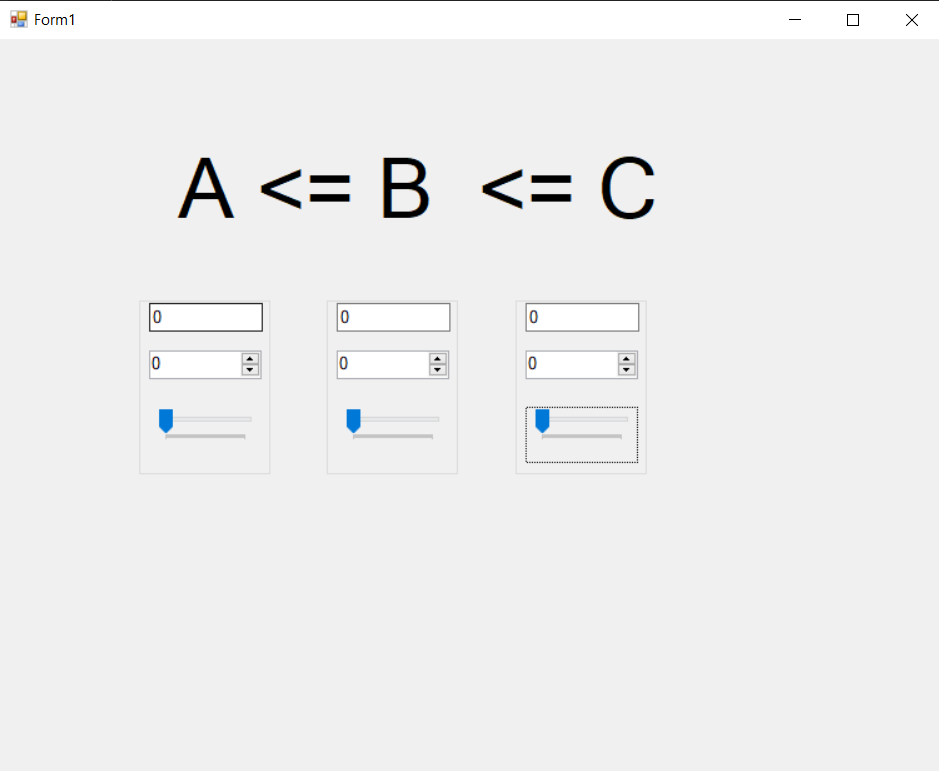


Рисунок . Внешний вид формы

Изменение поля С.

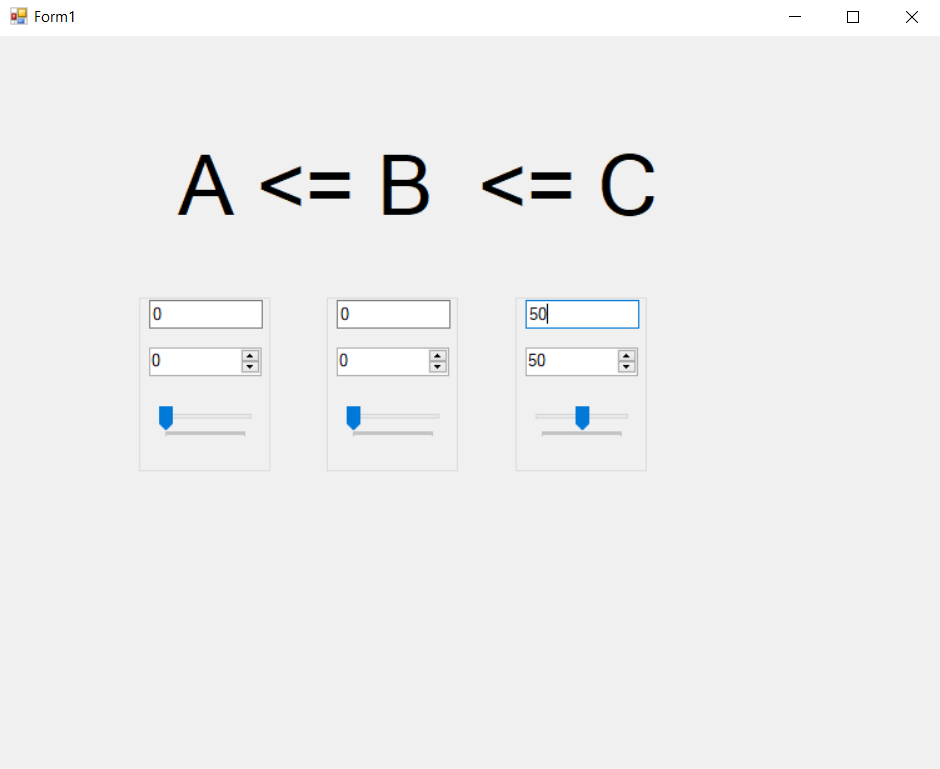


Рисунок . Изменение поля С

Изменение поля В.

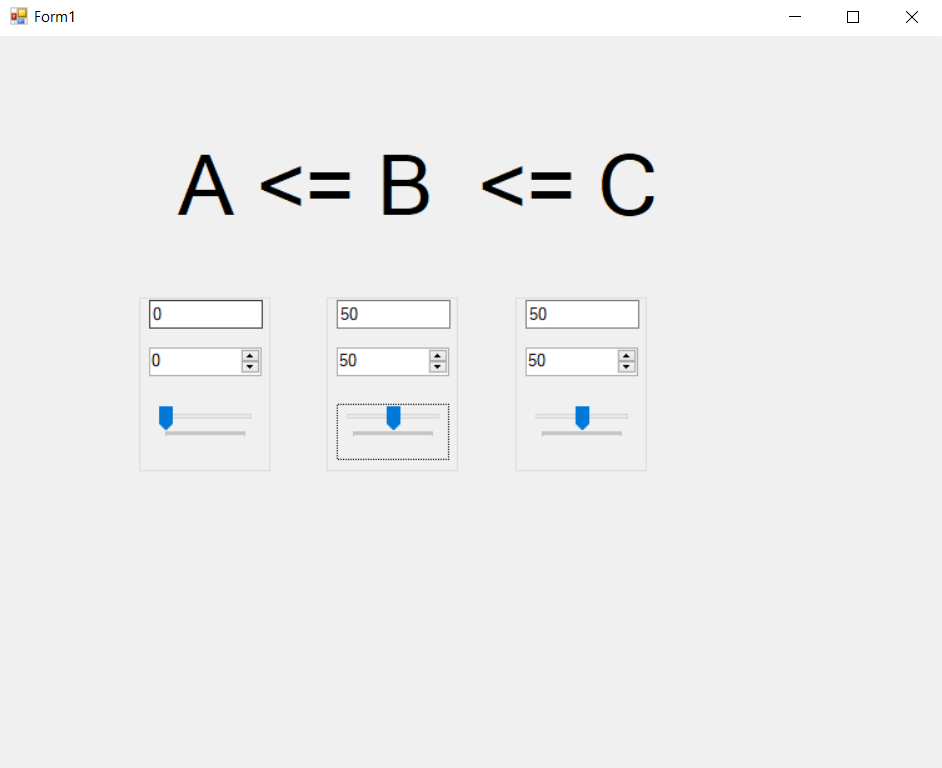


Рисунок . Изменение поля В.

Изменение поля А.

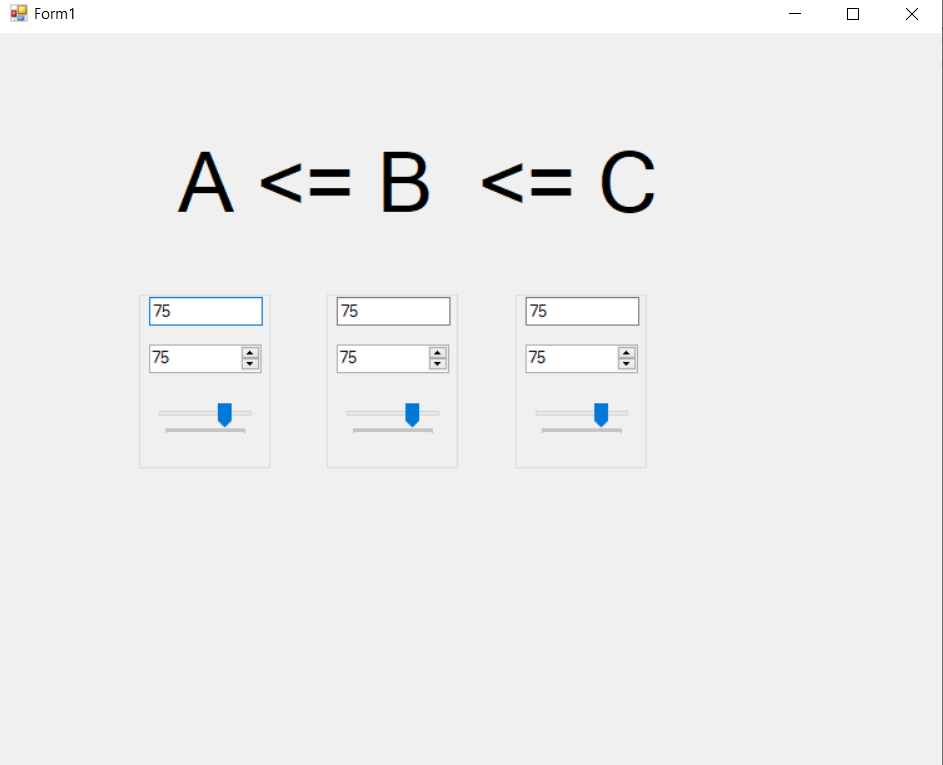


Рисунок . Изменение поля А

**Вывод**

В результате выполнения данной лабораторной работы было разработано MVC приложение, три числа и изменяющее их.

Полный код находится в Github:

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

**Приложение 1**

L4.2\_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;

using static L4.\_2\_OOP.Form1;

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

namespace L4.\_2\_OOP

{

public partial class Form1 : Form

{

Model model;

// Создание модели и добавление делегата

public Form1()

{

InitializeComponent();

model = new Model();

model.observers += new System.EventHandler(this.UpdateFromModel);

}

// Загрузка значений при открытии формы

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

{

model.setValue('a', Properties.Settings.Default.ModelCValue);

model.setValue('b', Properties.Settings.Default.ModelBValue);

model.setValue('c', Properties.Settings.Default.ModelAValue);

}

// Сохранение значений при закрытии формы

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

Properties.Settings.Default.ModelAValue = model.getValue('a');

Properties.Settings.Default.ModelBValue = model.getValue('b');

Properties.Settings.Default.ModelCValue = model.getValue('c');

Properties.Settings.Default.Save();

}

// Изменение значений

private void UpdateFromModel(object sender, EventArgs e)

{

tbA.Text = model.getValue('a').ToString();

nudA.Value = model.getValue('a');

trbA.Value = model.getValue('a');

tbB.Text = model.getValue('b').ToString();

nudB.Value = model.getValue('b');

trbB.Value = model.getValue('b');

tbC.Text = model.getValue('c').ToString();

nudC.Value = model.getValue('c');

trbC.Value = model.getValue('c');

}

// Класс модели

public class Model

{

private int valueA;

private int valueB;

private int valueC;

public System.EventHandler observers;

public void setValue(char symbol, int value)

{

if (value < 0)

value = 0;

else if (value > 100)

value = 100;

switch (symbol)

{

case 'a':

valueA = value;

if (valueA > valueB)

valueB = valueA;

if (valueA > valueC)

valueC = valueA;

break;

case 'b':

valueB = value;

if (valueB < valueA)

valueB = valueA;

if (valueB > valueC)

valueB = valueC;

break;

case 'c':

valueC = value;

if (valueC < valueA)

valueA = valueC;

if (valueC < valueB)

valueB = valueC;

break;

}

observers.Invoke(this, null);

}

public int getValue(char symbol)

{

switch (symbol)

{

case 'a':

return valueA;

case 'b':

return valueB;

case 'c':

return valueC;

}

return -1;

}

}

// Изменение значения в textBox-ах

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

{

if(e.KeyCode == Keys.Enter)

if(sender == tbA)

model.setValue('a', Convert.ToInt32(tbA.Text));

else if(sender == tbB)

model.setValue('b', Convert.ToInt32(tbB.Text));

else if (sender == tbC)

model.setValue('c', Convert.ToInt32(tbC.Text));

}

// Изменение значения в numericUpDown-ах

private void nud\_ValueChanged(object sender, EventArgs e)

{

if (sender == nudA)

model.setValue('a', Convert.ToInt32(nudA.Value));

else if (sender == nudB)

model.setValue('b', Convert.ToInt32(nudB.Value));

else if (sender == nudC)

model.setValue('c', Convert.ToInt32(nudC.Value));

}

// Изменение значения в TrackBar-ах

private void trb\_ValueChanged(object sender, EventArgs e)

{

if (sender == trbA)

model.setValue('a', Convert.ToInt32(trbA.Value));

else if (sender == trbB)

model.setValue('b', Convert.ToInt32(trbB.Value));

else if (sender == trbC)

model.setValue('c', Convert.ToInt32(trbC.Value));

}

}

}