Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

# Отчёт по программе «Индивидуальные задания по теме «Коллекции в с#»»

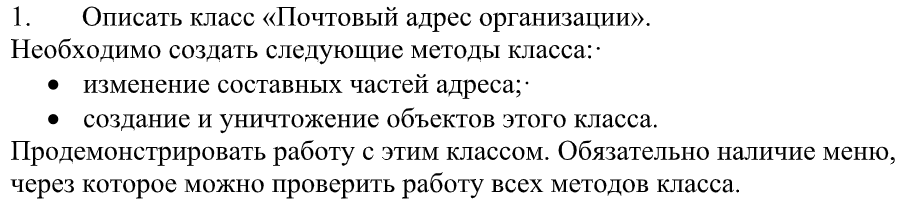
Выполнила: Форрат Лия

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2023

**Задание**



**Входные и выходные данные**

**string street - название улицы**

**string city-город**

**string state-страна**

**string code-индекс**

**Листинг программы**

*Основная программа*

int n = 1;

ArrayList al = new ArrayList();

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

{

n = 1;

if (new FileInfo("sd.txt").Length != 0)

{

StreamReader sr = File.OpenText("sd.txt");

al.Clear();

comboBox1.Items.Clear();

while (!sr.EndOfStream)

{

string[] s = sr.ReadLine().Split(',');

Addres a = new Addres();

a.State = s[0];

a.City = s[1];

a.Street = s[2];

a.Code = s[3];

listBox1.Items.Add(a.Getinfo());

al.Add(a);

comboBox1.Items.Add(n);

n++;

}

sr.Close();

}

else

{

try

{

Addres a1 = new Addres();

a1.State = textBox1.Text;

a1.City = textBox2.Text;

a1.Street = textBox3.Text;

a1.Code = textBox4.Text;

listBox1.Items.Add(a1.Getinfo());

comboBox1.Items.Add(n);

al.Add(a1);

n++;

StreamWriter sw = File.AppendText("sd.txt");

sw.WriteLine(a1.Getinfo());

sw.Close();

}

catch (FormatException)

{

MessageBox.Show("error", "error", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

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

{

al.RemoveAt(comboBox1.SelectedIndex);

comboBox1.Items.Remove(comboBox1.SelectedItem);

StreamWriter sw = File.CreateText("sd.txt");

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

{

Addres a2 = (Addres)al[i];

sw.WriteLine(a2.Getinfo());

}

sw.Close();

}

private void button5\_Click(object sender, EventArgs e)

{

listBox1.Items.Clear();

}

private void button3\_Click(object sender, EventArgs e)

{

try

{

Addres a1 = (Addres)al[comboBox1.SelectedIndex];

a1.Update(textBox1.Text, textBox2.Text, textBox3.Text, textBox4.Text);

al[comboBox1.SelectedIndex] = a1;

StreamWriter sw = File.CreateText("sd.txt");

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

{

Addres a2 = (Addres)al[i];

sw.WriteLine(a2.Getinfo());

}

sw.Close();

}

catch (FormatException) { MessageBox.Show("error", "error", MessageBoxButtons.OK, MessageBoxIcon.Error); }

}

private void button4\_Click(object sender, EventArgs e)

{

try

{

Addres a1 = new Addres();

a1.State = textBox1.Text;

a1.City = textBox2.Text;

a1.Street = textBox3.Text;

a1.Code = textBox4.Text;

listBox1.Items.Add(a1.Getinfo());

comboBox1.Items.Add(n);

al.Add(a1);

n++;

StreamWriter sw = File.CreateText("sd.txt");

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

{

Addres a2 = (Addres)al[i];

sw.WriteLine(a2.Getinfo());

}

sw.Close();

}

catch (FormatException)

{

MessageBox.Show("error", "error", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

*Класс*

private string street;

private string city;

private string state;

private string code;

public string Street

{

get { return street; }

set { street = value; }

}

public string City

{

get { return city; }

set { city = value; }

}

public string State

{

get { return state; }

set { state = value; }

}

public string Code

{

get { return code; }

set { code = value; }

}

public void Update(string state, string city, string street, string code)

{

this.street = street;

this.city = city;

this.state = state;

this.code = code;

}

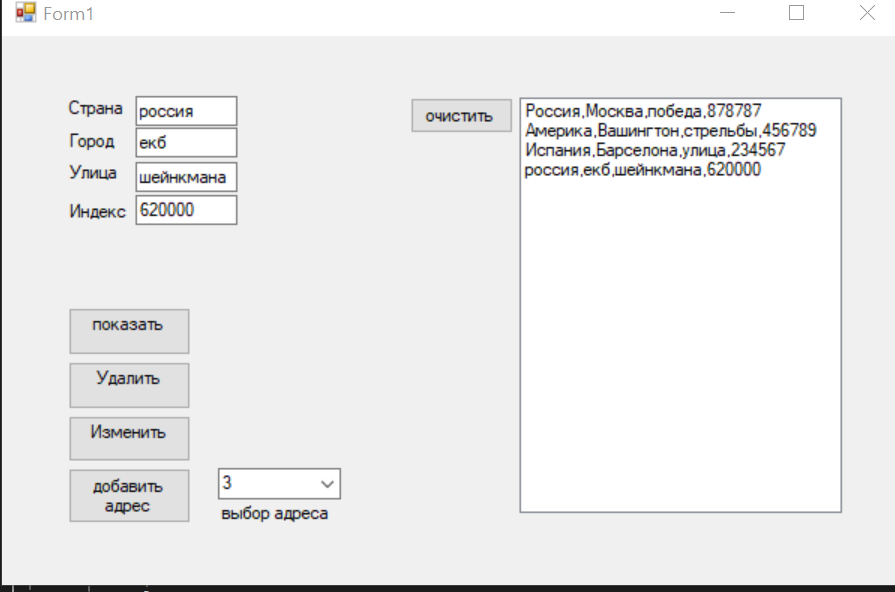
public string Getinfo()

{

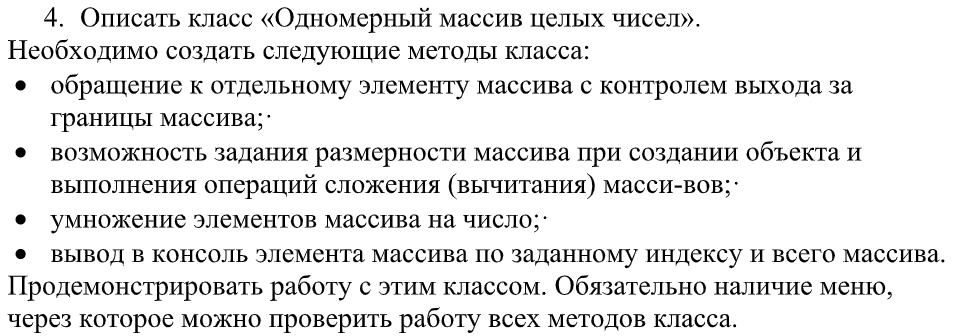
return $"{this.state},{this.city},{this.street},{this.code}";

}

**Тестовые ситуации**

****

**Задание**



**Входные данные**

**int size - размерность массива**

**int index- индекс элемента массива**

**int numb-число на которое умножается массив**

**string operation - опреция над массивом**

**string continues - выбор продолжить программу или нет**

**Выходные данные**

**выполненые операции над массивами**

**Листинг программы**

Основная программа

if (File.Exists("array.txt"))

{

Array array = new Array();

ArrayList arrayList = new ArrayList();

int size = 0, index = 0, numb = 0,j=0;

string operation = "", continues = "";

do

{

do

{

Console.Write("введите размер массива: ");

try

{

size = int.Parse(Console.ReadLine());

}

catch { Console.WriteLine("error"); }

}

while (size <= 0 || size > 20);

int[] mass = array.Size(size);

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

{

Console.Write($"{mass[i]} ");

}

StreamWriter r = File.CreateText("array.txt");

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

r.WriteLine(mass[i]);

r.Close();

do

{

Console.Write("\nвведите индекс элемента, чтобы его вывести (отсчет начинается с 1): ");

try

{

index = int.Parse(Console.ReadLine()) - 1;

if (index > mass.Length) Console.WriteLine("вы вышли за границы массива");

if (index < 0) Console.WriteLine("число не должно быть меньше 1!");

}

catch { Console.WriteLine("error"); }

}

while (index < 0|| index > mass.Length);

Console.WriteLine($"{array.Element(mass, index)}");

do

{

Console.Write("введите индекс элемента, чтобы начиная с него вывести массив (отсчет начинается с 1): ");

try

{

index = int.Parse(Console.ReadLine()) - 1;

if (index > mass.Length) Console.WriteLine("вы вышли за границы массива");

if (index < 0) Console.WriteLine("число не должно быть меньше 1!");

}

catch { Console.WriteLine("error"); }

}

while (index < 0 || index > mass.Length);

int[] sup = array.Out(mass, index);

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

{

Console.Write($"{sup[i]} ");

}

try

{

Console.Write("\nвведите число, на которое нужно умножить массив: ");

numb = int.Parse(Console.ReadLine());

}

catch (FormatException)

{

Console.WriteLine("error");

}

int[] sup2 = array.Multiply(mass, numb);

for (int i = 0; i < sup2.Count(); i++)

{

Console.Write($"{sup2[i]} ");

}

do

{

Console.Write("\nвыберите операцию: + или -: ");

operation = Convert.ToString(Console.ReadLine());

}

while (!(operation == "+" || operation == "-"));

int[] mass2 = array.Size(mass.Length);

int[] massev = new int[mass.Length];

StreamReader sr = File.OpenText("array.txt");

while (!sr.EndOfStream)

{

massev[j] = Convert.ToInt32(sr.ReadLine());

j++;

}

sr.Close();

Console.Write("первый массив: ");

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

{

Console.Write($"{massev[i]} ");

}

Console.Write("\nвторой массив: ");

for (int i = 0; i < mass2.Length; i++)

{

Console.Write($"{mass2[i]} ");

}

arrayList = array.Operation(massev, mass2, operation);

Console.Write("\nрезультат: ");

foreach (object action in arrayList)

{

Console.Write($"{action} ");

}

Console.Write("\nчтобы продолжить выполнение программы напишите 'X', чтобы завершить 'Y':");

continues = Convert.ToString(Console.ReadLine());

}

while (continues == "X");

if (continues == "Y") Console.WriteLine("выполнение завершено");

}

else Console.WriteLine("файл 'array.txt' не найден!");

Console.ReadKey();

Класс

private int[] array;

Random rnd = new Random();

public int[] Size(int size)

{

array = new int[size];

for (int i = 0; i < size; i++)

{

array[i] = rnd.Next(-100,100);

}

return array;

}

public string Number(int index)

{

if (index >= array.Length || index < 0) return $"нет элемента с такой позицией";

else return $"результат: {array[index]}";

}

public int Element (int[] mas, int index)

{

return mas[index];

}

public int[] Out(int[] mas, int index)

{

int count = 0;

for (int i = index; i < mas.Count(); i++)

{

count++;

}

int[] mas2 = new int[count];

int j = 0;

for (int i = index; i < mas.Count(); i++)

{

mas2[j] = mas[i];

j++;

}

return mas2;

}

public int[] Multiply(int[] mas, int mult)

{

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

{

mas[i] \*= mult;

}

return mas;

}

public ArrayList Operation (int[] mas1, int[] mas2, string answer)

{

ArrayList arrayList = new ArrayList();

if(answer=="+")

for (int i = 0; i < mas1.Count(); i++)

{

arrayList.Add(mas1[i] + mas2[i]);

}

else

for (int i = 0; i < mas1.Count(); i++)

{

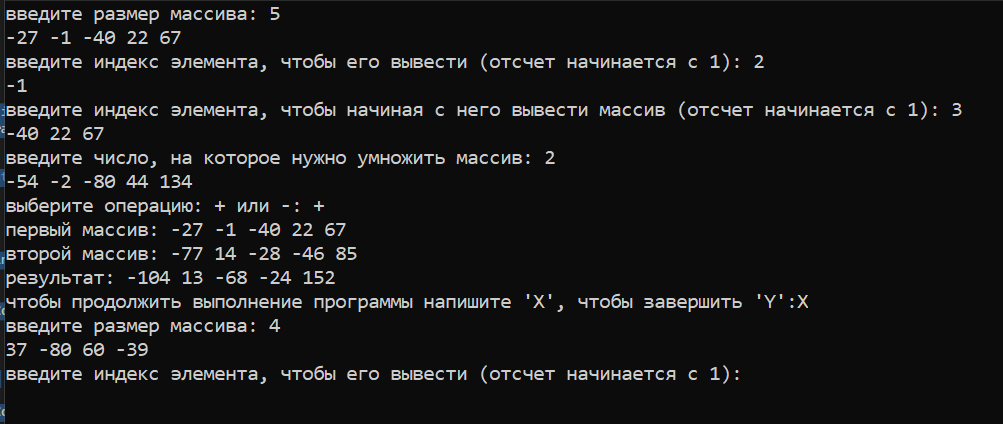
arrayList.Add(mas1[i] - mas2[i]);

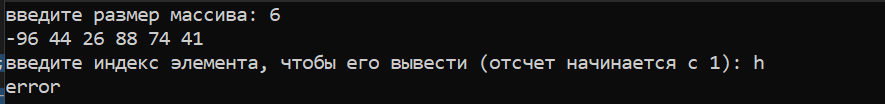
}

return arrayList;

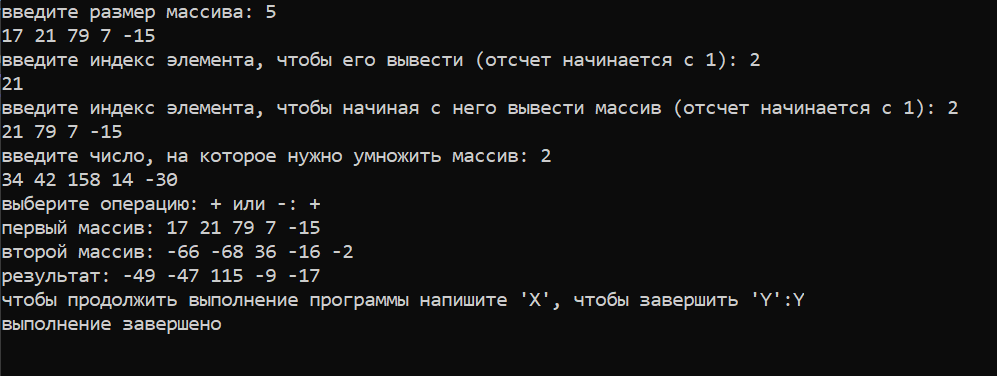
}

**Тестовые ситуации**

****

****

****

****