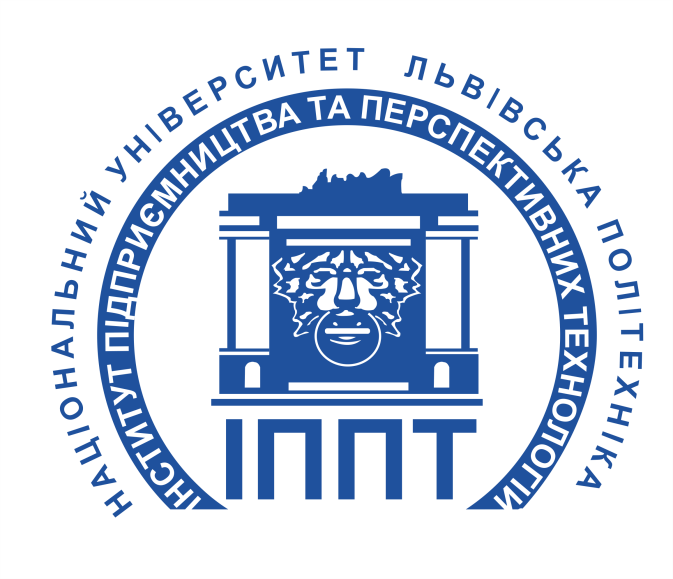
**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Н-Н ІНСТИТУТ ПІДПРИЄМНИЦТВА ТА ПЕРСПЕКТИВНИХ ТЕХНОЛОГІЙ**



**ЗВІТ**

до виконаної лабораторної роботи №4

на тему

***«*перевантаження операторів для роботи з багатовимірними масивами*»***

Виконав:

студент гр. КН-214

**Дубницький Ю.І.**

Прийняла:

**Машевська М.В.**

**Львів–2018**

**Лаболаторна робота №4**

**Завдання**

Згенерувати 2 масиви (*А* і *В*) чисел з інтервалу [0, 10].

Забезпечити можливість (**розробити метод**) дублювання елементів рядка (або стовпця) *i* масиву *А* в рядок (або стовпець) *j* масиву *В*.

Розмірність масиву задається користувачем: 3×3, або 4×4, або 5×5 (використати елемент *domainUpDown*). Розмірність масивів є однаковою.

**Розробити** **метод**, що дозволяє отримати елементи рядка або стовпця матриці, у вигляді одновимірного масиву чисел.

**Розробити клас** Одновимірний­\_Масив. Для цього класу:

• перевизначити оператор "+", що дозволить сумувати елементи двох одновимірних масивів;

• перевизначити оператор "/", що дозволить знайти цілу частину від ділення елементів одновимірного масиву на число *с*;

• перевизначити оператор "++", що дозволить збільшити елементи одновимірного масиву на 2;

• перевизначити оператор "==", що дозволить перевіряти рівність елементів двох одновимірних масивів (не забути, що оператор "==" необхідно перевантажувати разом з оператором "!=").

**В основній програмі** (використовуючи розроблений клас з перевизначеними операторами) **знайти**:

• суму заданих рядків матриці А та матриці В (результатом є одновимірний масив сум);

• частку від ділення елементів сповпця *k* масиву В на число *с*, введеного користувачем (елемент numericUpDown [min=2, max=4]), (результатом є одновимірний масив);;

• інкрементувати (оператор "++") елементи рядка *p* масиву А (результатом є одновимірний масив);

• перевірити рівність рядків *m* матриці *А* та *n* матриці *В* (результатом є твердження про рівність).

Індекси рядків та стовпців для роботи задаються в елементах (numericUpDown [min=1, max – залежить від розмірності матриць]).

Результати (одновимірні масиви) виводяться в елементах textBox.

Результат перевірки рівності елементів вивести за допомогою messageBox.

Використати, зокрема, такі елементи графічного інтерфейсу: listBox, numericUpDown, toolStrip, label.

**Код програми**

**Form1.cs**

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 laboratorna4c

{

public partial class Form1 : Form

{

static int[,] arrA;

static int[,] arrB;

private int[,] ArrayGeneration(int z)

{

Random rand = new Random();

int[,] arr = new int[z,z];

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

{

for (int j = 0; j < z; j++)

{

arr[i,j] = rand.Next(0, 10);

}

}

return arr;

}

private void ArrayOutput(int[,] array,int variant , int numb)

{

string output = "";

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

{

output = "";

for (int j = 0; j < numb; j++)

{

output += array[i,j] + " ";

}

if (variant == 1)

listBox1.Items.Add(output);

if (variant == 2)

listBox2.Items.Add(output);

}

}

private void Incremention(int[,] array , int numb)

{

int ads;

int number = comboBox1.SelectedIndex;

for (int i = 0; i < number+3; i++) {

ads = array[numb, i];

ads = ads \* 2;

textBox2.Text += (ads+" ");

}

}

private int Summa(int[,] array , int numb)

{

int suuma = 0;

int number = comboBox1.SelectedIndex;

for (int i = 0; i < number + 3; i++) {

suuma += array[numb, i];

}

return suuma;

}

private void Qwerty(int[,] arrayA , int[,] arrayB, int a, int b)

{

int number = comboBox1.SelectedIndex;

int suum = number + 3;

for (int i = 0; i < number + 3; i++) {

if (Summa(arrA, a - 1) == Summa(arrB, b-1)) {

MessageBox.Show("рядки рівні");

}

MessageBox.Show("No no no ");

break;

}}

public Form1()

{

InitializeComponent();

}

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

{

object number = comboBox1.SelectedIndex;

int abs = 0;

switch (number)

{

case 0:

abs = 3;

break;

case 1:

abs = 4;

break;

case 2:

abs = 5;

break;

default:

break;

}

listBox1.Items.Clear();

arrA = ArrayGeneration(abs);

ArrayOutput(arrA, 1, abs);

}

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

{

object number = comboBox1.SelectedIndex;

int abs = 0;

switch (number)

{

case 0:

abs = 3;

break;

case 1:

abs = 4;

break;

case 2:

abs = 5;

break;

default:

break;

}

listBox2.Items.Clear();

arrB = ArrayGeneration(abs);

ArrayOutput(arrB, 2, abs);

}

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

{

textBox2.Text = "";

int number = Convert.ToInt16(numericUpDown7.Value)-1;

//textBox2.Text = ("");

//Incremention(arrA, number);

Yomayo mas\_a = new Yomayo(arrA, number);

++mas\_a;

foreach (int x in mas\_a.arr)

textBox2.Text += x.ToString() + " ";

}

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

{

int numberA = Convert.ToInt16(numericUpDown3.Value) -1;

int numberB = Convert.ToInt16(numericUpDown4.Value) -1;

int[] result = new int[numberA];

Yomayo mas\_a = new Yomayo(arrA, numberA);

Yomayo mas\_b = new Yomayo(arrB, numberB);

result = mas\_a + mas\_b;

foreach (int x in result)

listBox3.Items.Add( x.ToString() + " ");

//listBox3.Items.Add(Summa(arrA, numberA-1) + Summa(arrB, numberB-1));

}

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

{

listBox3.Items.Clear();

}

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

{

toolStripLabel1.Text = ("рядок");

toolStripLabel5.Text = ("рядок");

int numb1 = Convert.ToInt16(toolStripTextBox1.Text);

int numb2 = Convert.ToInt16(toolStripTextBox2.Text);

int adsd;

int number = comboBox1.SelectedIndex;

listBox2.Items.Clear();

for (int i = 0; i < number + 3; i++)

{

adsd = arrA[numb1 - 1, i];

arrB[numb2 - 1, i] = adsd;

}

int abs = 0;

switch (number)

{

case 0:

abs = 3;

break;

case 1:

abs = 4;

break;

case 2:

abs = 5;

break;

default:

break;

}

ArrayOutput(arrB, 2, abs);

}

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

{

toolStripLabel1.Text = ("стопчик");

toolStripLabel5.Text = ("стопчик");

int numb1 = Convert.ToInt16(toolStripTextBox1.Text);

int numb2 = Convert.ToInt16(toolStripTextBox2.Text);

int adsd;

int number = comboBox1.SelectedIndex;

listBox2.Items.Clear();

for (int i = 0; i < number + 3; i++)

{

adsd = arrA[i, numb1-1];

arrB[i,numb2 - 1] = adsd;

}

int abs = 0;

switch (number)

{

case 0:

abs = 3;

break;

case 1:

abs = 4;

break;

case 2:

abs = 5;

break;

default:

break;

}

ArrayOutput(arrB, 2, abs);

}

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

{

textBox1.Text = ("");

int numberA = Convert.ToInt16(numericUpDown5.Value)-1;

int numberB = Convert.ToInt16(numericUpDown6.Value);

int number = comboBox1.SelectedIndex;

//int adsd;

//for (int i = 0; i < number + 3; i++)

//{

// adsd = arrB[i, numberA - 1];

// textBox1.Text += (adsd % numberB + " ");

//}

int[] result = new int[numberA];

Yomayo mas\_a = new Yomayo(arrB, numberA);

result = mas\_a % numberB;

foreach (int x in result)

textBox1.Text += x.ToString() + " ";

}

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

{

int numberA = Convert.ToInt16(numericUpDown1.Value);

int numberB = Convert.ToInt16(numericUpDown2.Value);

bool result;

Yomayo mas\_a = new Yomayo(arrA, numberA);

Yomayo mas\_b = new Yomayo(arrB, numberB);

result = mas\_a == mas\_b;

if (result)

MessageBox.Show("Рядки рівні");

else

MessageBox.Show("Рядки не рівні");

//Qwerty(arrA, arrB, numberA, numberB);

}

}

class Yomayo

{

public int[] arr;

public Yomayo()

{

this.arr = new int[1];

}

public Yomayo(int[] arr)

{

this.arr = arr;

}

public Yomayo(int[,] arr, int row)

{

int[] mas = new int[arr.GetLength(1)];

for (int i = 0; i < arr.GetLength(1); i++)

mas[i] = arr[row, i];

this.arr = mas;

}

public static int[] operator +(Yomayo a, Yomayo b)

{

int[] mas = new int[a.arr.Length];

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

{

mas[i] = a.arr[i] + b.arr[i];

}

return mas;

}

public static int[] operator %(Yomayo a, int b)

{

int[] mas = new int[a.arr.Length];

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

{

mas[i] = a.arr[i] % b;

}

return mas;

}

public static Yomayo operator ++(Yomayo a)

{

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

{

a.arr[i] = a.arr[i] + 2;

}

return a;

}

public static bool operator ==(Yomayo a, Yomayo b)

{

int count = 0;

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

{

if (a.arr[i] == b.arr[i])

count += 1;

}

if (count == a.arr.Length)

return true;

else

return false;

}

public static bool operator !=(Yomayo a, Yomayo b)

{

int count = 0;

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

{

if (a.arr[i] == b.arr[i])

count += 1;

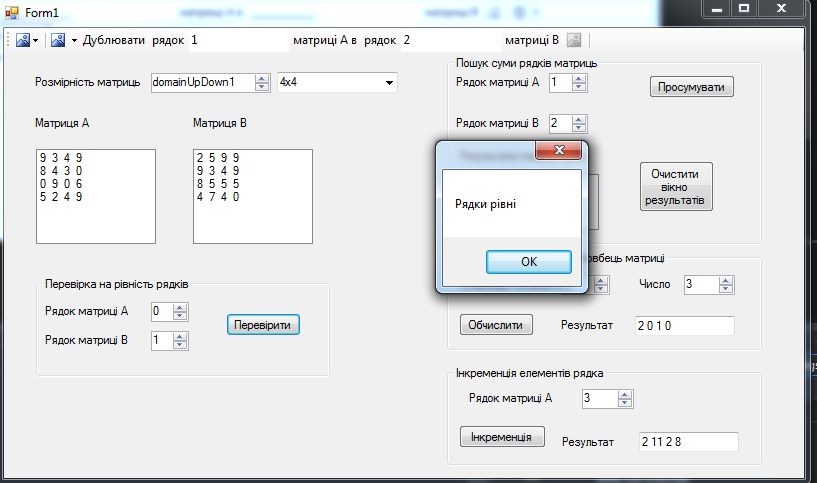
}

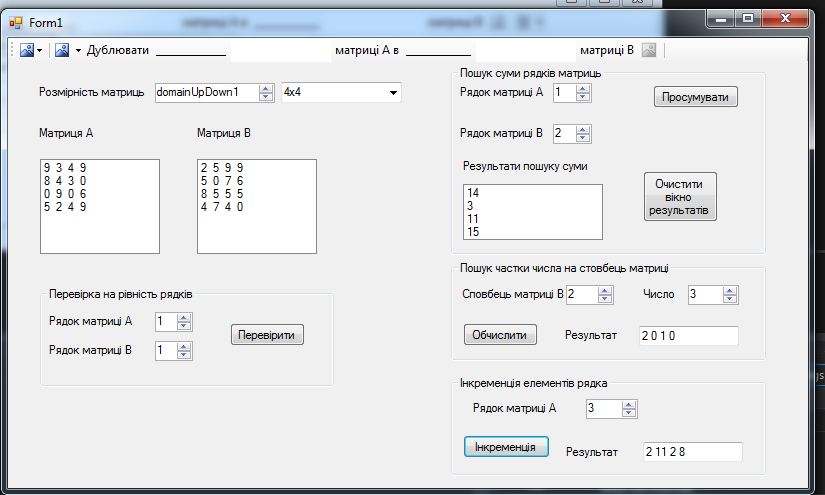
if (count == a.arr.Length)

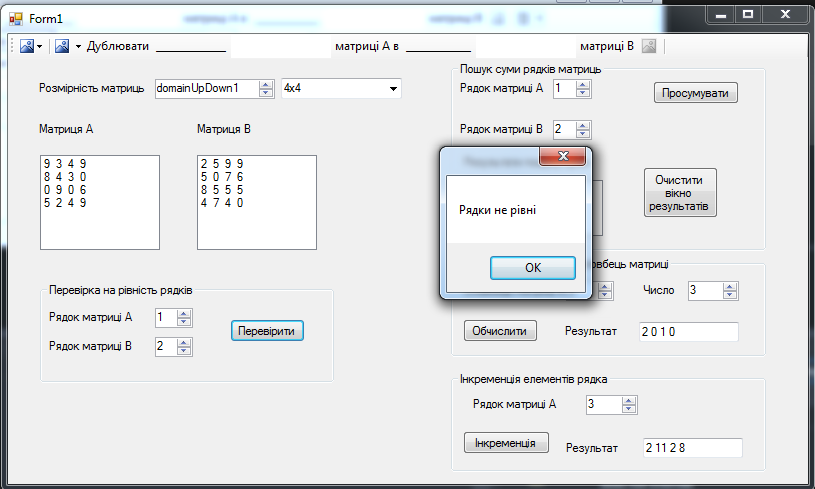
return false;

else

return true;} };







**Висновок:** У цій лаболаторній роботі я розробив програму в Form Application для перевантаження операторів для роботи з багатовимірними масивами.