Чернівецький національний університет імені Юрія Федьковича

Навчально-науковий інститут фізико-технічних та комп’ютерних наук

Кафедра математичних проблем управління і кібернетики

**Звіт**

про виконання лабораторної роботи №2

«Цикли. Масиви.
»

з дисципліни

«Крос-платформне програмування»

Виконав: студент 241 групи

Фрасинюк О.Б.

Перевірив: канд.фіз.-мат. наук,

доцент Лазорик В.В.

Оцінка:

Дата захисту:

Чернівці 2024

Варіант 17

<https://github.com/VLazorykOOP/lab2csarp25v2-Oleh-Frasyniuk>

Задано масив

1.17. Визначити, чи є добуток елементів тризначним числом.

**При вирішенні завдання використовувати одновимірний масив.**

**Розмірність масиву вводиться з клавіатури.**

using System;

class Program

{

static void Main()

{

Console.Write("Введіть розмірність масиву: ");

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

double[] array = new double[n];

Console.WriteLine("Введіть елементи масиву:");

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

{

array[i] = double.Parse(Console.ReadLine());

}

double product = 1;

foreach (double num in array)

{

product \*= num;

}

if (Math.Abs(product) >= 100 && Math.Abs(product) < 1000)

{

Console.WriteLine("Добуток є тризначним числом.");

}

else

{

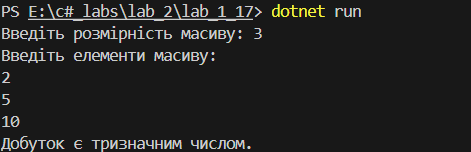
Console.WriteLine("Добуток не є тризначним числом.");

}

}

}

Результат:



Завдання 2. Дана послідовність з n дійсних чисел

2.17. Підрахувати кількість елементів, значення яких більше значення

попереднього елемента.

**При вирішенні завдання використовувати двовимірний масив.**

using System;

class Program

{

static void Main()

{

Console.Write("Введіть кількість рядків (n): ");

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

Console.Write("Введіть кількість стовпців (m): ");

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

double[,] array = new double[n, m];

Console.WriteLine("Введіть елементи масиву:");

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

{

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

{

array[i, j] = double.Parse(Console.ReadLine());

}

}

int count = 0;

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

{

for (int j = 1; j < m; j++) // Починаємо з 1, щоб мати попередній елемент

{

if (array[i, j] > array[i, j - 1])

{

count++;

}

}

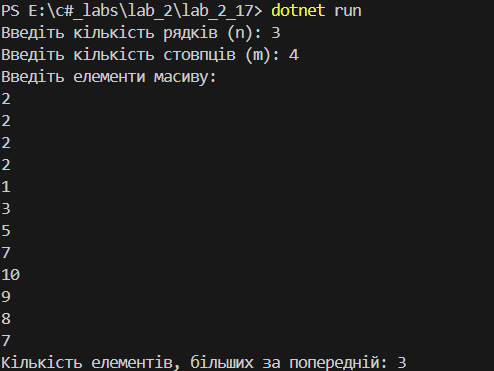
}

Console.WriteLine($"Кількість елементів, більших за попередній: {count}");

}

}

Результат:



Завдання 3. Дано масив розміром n×n, елементи якого цілі числа.

3.2. Підрахувати суму елементів, розташованих на побічної діагоналі.

**Завдання з масивами вирішити двома способами, використовуючи одновимірний масив, а потім двовимірний. Розмірність масиву вводиться з клавіатури.**

using System;

class Program

{

static void Main()

{

Console.Write("Введіть розмірність квадратного масиву (n): ");

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

int[,] matrix = new int[n, n];

Console.WriteLine("Введіть елементи двовимірного масиву:");

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

{

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

{

matrix[i, j] = int.Parse(Console.ReadLine());

}

}

// Підрахунок суми побічної діагоналі (двовимірний масив)

int sum2D = 0;

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

{

sum2D += matrix[i, n - 1 - i];

}

Console.WriteLine($"Сума елементів побічної діагоналі (2D масив): {sum2D}");

// Одновимірне представлення масиву

int[] oneDArray = new int[n \* n];

for (int i = 0, k = 0; i < n; i++)

{

for (int j = 0; j < n; j++, k++)

{

oneDArray[k] = matrix[i, j];

}

}

// Підрахунок суми побічної діагоналі (одновимірний масив)

int sum1D = 0;

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

{

sum1D += oneDArray[i \* n + (n - 1 - i)];

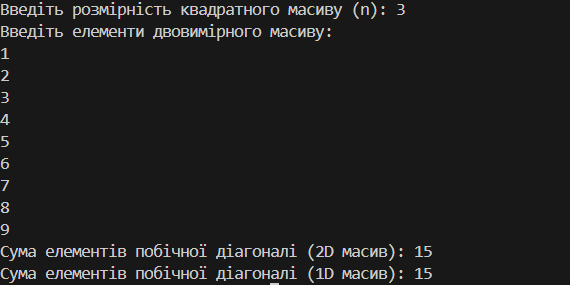
}

Console.WriteLine($"Сума елементів побічної діагоналі (1D масив): {sum1D}");

}

}

Результат:



Завдання 4. Дано східчастий масив з n рядків, у рядках по mj (j=1..n) елементів.   
 4.2. Для кожного рядка підрахувати кількість додатних елементів і записати дані в новий масив.

**Завдання з масивами вирішити двома способами, використовуючи одновимірний масив, а потім двовимірний. Розмірність масиву вводиться з клавіатури.**

using System;

class Program

{

static void Main()

{

Console.Write("Введіть кількість рядків (n): ");

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

int[][] jaggedArray = new int[n][];

// Заповнення східчастого масиву

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

{

Console.Write($"Введіть кількість елементів у рядку {i + 1}: ");

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

jaggedArray[i] = new int[m];

Console.WriteLine($"Введіть {m} елементів рядка {i + 1}:");

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

{

jaggedArray[i][j] = int.Parse(Console.ReadLine());

}

}

// Підрахунок кількості додатних елементів у кожному рядку

int[] positiveCounts = new int[n];

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

{

int count = 0;

foreach (int num in jaggedArray[i])

{

if (num > 0) count++;

}

positiveCounts[i] = count;

}

Console.WriteLine("Кількість додатних елементів у кожному рядку:");

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

{

Console.WriteLine($"Рядок {i + 1}: {positiveCounts[i]}");

}

// Двовимірний масив

int maxColumns = 0;

foreach (var row in jaggedArray)

{

if (row.Length > maxColumns) maxColumns = row.Length;

}

int[,] twoDArray = new int[n, maxColumns];

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

{

for (int j = 0; j < jaggedArray[i].Length; j++)

{

twoDArray[i, j] = jaggedArray[i][j];

}

}

// Підрахунок для двовимірного масиву

int[] positiveCounts2D = new int[n];

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

{

int count = 0;

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

{

if (twoDArray[i, j] > 0) count++;

}

positiveCounts2D[i] = count;

}

Console.WriteLine("Кількість додатних елементів (двовимірний масив):");

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

{

Console.WriteLine($"Рядок {i + 1}: {positiveCounts2D[i]}");

}

}

}

Результат:

