ст. Гр КН-923с

Кузьмін Станіслав

**Звіт з лабораторної роботи №2**

Використання масивів і колекцій у мові C# .

Мета роботи: набути навички в написанні програм з використання масивів і колекцій.

Завдання 1 Використовуючи завдання власного варіанта до лабораторної роботи 8 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

/\*Треба оголосити масив цілих чисел і заповнити його випадковими значеннями.

\* Розмір масиву і діапазон значень його елементів задані у виданому варіанті індивідуального завдання (див. п.6).

\* В індивідуальних завданнях вказано також, яку обробку масиву слід провести. \*/

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

int[] Ar = new int[100];

Random rand = new Random();

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

{

Ar[i] = rand.Next(-100, 100);

}

int currentSum = 0;

int maxSum = 0;

int start = -1;

int maxStart = -1;

int maxEnd = -1;

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

{

if (Ar[i] > 0)

{

if (currentSum == 0)

{

start = i;

}

currentSum += Ar[i];

if (currentSum > maxSum)

{

maxSum = currentSum;

maxStart = start;

maxEnd = i;

}

}

else

{

currentSum = 0;

}

}

Console.WriteLine("Масив : ");

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

{

Console.Write($"{Ar[i],5}");

}

Console.WriteLine("\n\nПослідовність з максимальною сумою:");

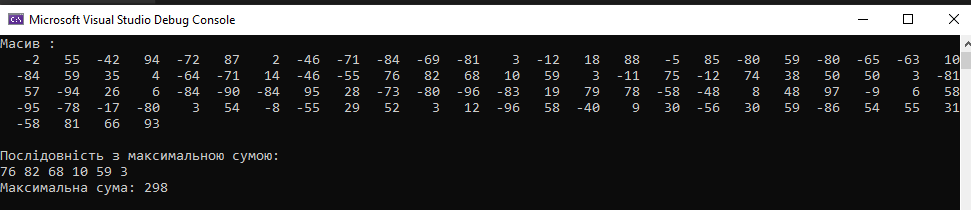
for (int i = maxStart; i <= maxEnd; i++)

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

Console.WriteLine($"\nМаксимальна сума: {maxSum}");

}

}



Завдання 2\* Виконати минуле завдання, зробивши похідний масив динамічним, задавши його розмір з консолі.

//Виконати минуле завдання, зробивши похідний масив динамічним, задавши

//його розмір з консолі.

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

int size;

Console.Write("Введіть розмір масиву - ");

size = Convert.ToInt32(Console.ReadLine());

int[] Ar = new int[size];

Random rand = new Random();

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

{

Ar[i] = rand.Next(-100, 100);

}

int currentSum = 0;

int maxSum = 0;

int start = -1;

int maxStart = -1;

int maxEnd = -1;

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

{

int value = Ar[i];

if (value > 0)

{

if (currentSum == 0)

{

start = i;

}

currentSum += value;

if (currentSum > maxSum)

{

maxSum = currentSum;

maxStart = start;

maxEnd = i;

}

}

else

{

currentSum = 0;

}

}

Console.WriteLine("Масив : ");

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

{

Console.Write($"{Ar[i],5} ");

}

Console.WriteLine("\n\nПослідовність з максимальною сумою:");

for (int i = maxStart; i <= maxEnd; i++)

{

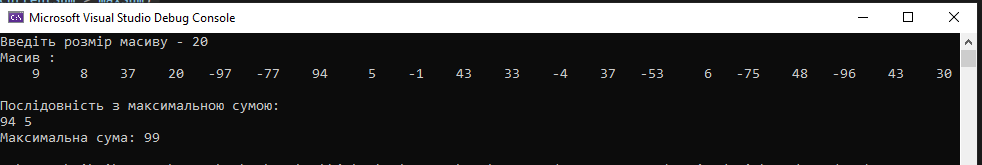
Console.Write($"{Ar[i]} ");

}

Console.WriteLine($"\nМаксимальна сума: {maxSum}");

}

}



Завдання 3\* Виконати минуле завдання, використовуючи кольорові можливості консолі для покращення інтерфейсу роботи програми. Наприклад, виділяти різними кольорами знайдені в масивах за завданням елементи чи їхні послідовності, розраховані параметри, тощо.

//Виконати минуле завдання, використовуючи кольорові можливості консолі

//для покращення інтерфейсу роботи програми. Наприклад, виділяти різними

//кольорами знайдені в масивах за завданням елементи чи їхні послідовності,

//розраховані параметри, тощо.

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

int size;

Console.ForegroundColor = ConsoleColor.Green;

Console.Write("Введіть розмір масиву - ");

size = Convert.ToInt32(Console.ReadLine());

int[] Ar = new int[size];

Random rand = new Random();

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

{

Ar[i] = rand.Next(-100, 100);

}

int currentSum = 0;

int maxSum = 0;

int start = -1;

int maxStart = -1;

int maxEnd = -1;

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

{

int value = Ar[i];

if (value > 0)

{

if (currentSum == 0)

{

start = i;

}

currentSum += value;

if (currentSum > maxSum)

{

maxSum = currentSum;

maxStart = start;

maxEnd = i;

}

}

else

{

currentSum = 0;

}

}

Console.ForegroundColor = ConsoleColor.White;

Console.WriteLine("Масив : ");

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

{

int value = Ar[i];

if (value > 0)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.Write($"{value,5}");

}

else

{

Console.ForegroundColor = ConsoleColor.Blue;

Console.Write($"{value,5}");

}

}

Console.ForegroundColor = ConsoleColor.Yellow;

Console.WriteLine("\n\nПослідовність з максимальною сумою:");

for (int i = maxStart; i <= maxEnd; i++)

{

int value = Ar[i];

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

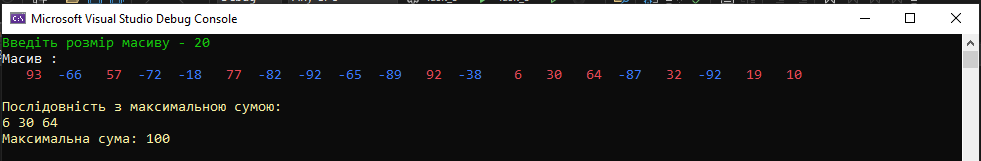
}

Console.WriteLine($"\nМаксимальна сума: {maxSum}");

Console.ResetColor();

}

}



Завдання 4\* Виконати минуле завдання, використовуючи замість масивів одну з колекцій мови C# (List, LinkedList, ArrayList, чи ін.).

//Виконати минуле завдання, використовуючи замість масивів одну з колекцій

//мови C# (List,

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

int size;

Console.ForegroundColor = ConsoleColor.Green;

Console.Write("Введіть розмір масиву - ");

size = Convert.ToInt32(Console.ReadLine());

List<int> Ar = new List<int>(size);

Random rand = new Random();

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

{

Ar.Add(rand.Next(-100, 100));

}

int currentSum = 0;

int maxSum = 0;

int start = -1;

int maxStart = -1;

int maxEnd = -1;

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

{

if (Ar[i] >= 0)

{

if (currentSum == 0)

{

start = i;

}

currentSum += Ar[i];

if (currentSum > maxSum)

{

maxSum = currentSum;

maxStart = start;

maxEnd = i;

}

}

else

{

currentSum = 0;

}

}

Console.ForegroundColor = ConsoleColor.White;

Console.WriteLine("Масив : ");

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

{

if (Ar[i] > 0)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.Write($"{Ar[i],5}");

}

else

{

Console.ForegroundColor = ConsoleColor.Blue;

Console.Write($"{Ar[i],5}");

}

//Console.Write($"{Ar[i],5}");

}

Console.ForegroundColor = ConsoleColor.Yellow;

Console.WriteLine("\n\nПослідовність з максимальною сумою:");

for (int i = maxStart; i <= maxEnd; i++)

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

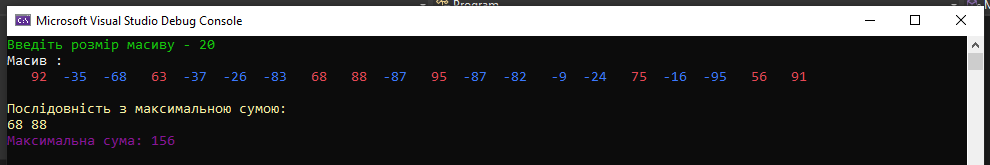
Console.ForegroundColor = ConsoleColor.DarkMagenta;

Console.WriteLine($"\nМаксимальна сума: {maxSum}");

Console.ResetColor();

}

}



Завдання 5 Використовуючи завдання власного варіанта до лабораторної роботи 9 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

/\* Створити квадратну матрицю розміром 9х9.

\* В індивідуальних завданнях (див. п.6) вказано,

\* який тип даних використати та яку обробку матриці потрібно виконати. \*/

using System;

class Program

{

const int S = 9;

static void Main()

{

int[,] Ar = new int[S, S];

int k = 1;

for (int start = 0; start < S; start++)

{

int row = start;

int col = 0;

while (row >= 0)

{

Ar[row, col] = k;

k++;

row--;

col++;

}

}

for (int start = 1; start < S; start++)

{

int row = S - 1;

int col = start;

while (col < S)

{

Ar[row, col] = k;

k++;

row--;

col++;

}

}

for (int l = 0; l < S; l++)

{

for (int r = 0; r < S; r++)

{

Console.Write($"{Ar[l, r],4} ");

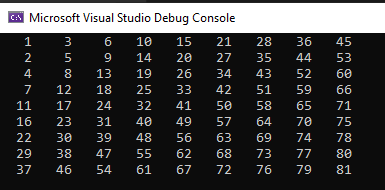
}

Console.WriteLine();

}

}

}



Завдання 6\* Виконати минуле завдання, зробивши матрицю динамічною, задавши її розміри з консолі, використовуючи для цього прямокутний ([ , ]) двовимірний масив

//Виконати минуле завдання, зробивши матрицю динамічною, задавши її

//розміри з консолі, використовуючи для цього прямокутний ([ ,])

//двовимірний масив

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

Console.Write("Введіть розмір матриці: ");

int S = Convert.ToInt32(Console.ReadLine());

Console.WriteLine($"Розмір: {S}");

int[,] Ar = new int[S, S];

int k = 1;

for (int start = 0; start < S; start++)

{

int row = start;

int col = 0;

while (row >= 0)

{

Ar[row, col] = k;

k++;

row--;

col++;

}

}

for (int start = 1; start < S; start++)

{

int row = S - 1;

int col = start;

while (col < S)

{

Ar[row, col] = k;

k++;

row--;

col++;

}

}

Console.WriteLine("Результат:");

for (int l = 0; l < S; l++)

{

for (int r = 0; r < S; r++)

{

Console.Write($"{Ar[l, r],4} ");

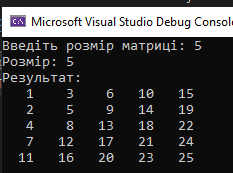
}

Console.WriteLine();

}

}

}



Завдання 7\* Виконати минуле завдання, зробивши матрицю динамічною, задавши її розміри з консолі, використовуючи для цього зубчастий ([][]) двовимірний масив.

//Виконати минуле завдання, зробивши матрицю динамічною, задавши її

//розміри з консолі, використовуючи для цього зубчастий ([][]) двовимірний

//масив.

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

Console.Write("Введіть розмір матриці: ");

int S = Convert.ToInt32(Console.ReadLine());

Console.WriteLine($"Розмір: {S}");

int[][] Ar = new int[S][];

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

{

Ar[i] = new int[S];

}

int k = 1;

for (int start = 0; start < S; start++)

{

int row = start;

int col = 0;

while (row >= 0)

{

Ar[row][col] = k;

k++;

row--;

col++;

}

}

for (int start = 1; start < S; start++)

{

int row = S - 1;

int col = start;

while (col < S)

{

Ar[row][col] = k;

k++;

row--;

col++;

}

}

Console.WriteLine("Результат:");

for (int l = 0; l < S; l++)

{

for (int r = 0; r < S; r++)

{

Console.Write($"{Ar[l][r],4} ");

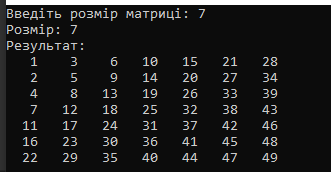
}

Console.WriteLine();

}

}

}



Завдання 8\* Виконати минуле завдання, використовуючи кольорові можливості консолі для покращення інтерфейсу роботи програми. Наприклад, виділяти різними кольорами знайдені в матриці за завданням елементи чи їхні послідовності, розраховані параметри, тощо.

//Виконати минуле завдання, використовуючи кольорові можливості консолі

//для покращення інтерфейсу роботи програми. Наприклад, виділяти різними

//кольорами знайдені в матриці за завданням елементи чи їхні послідовності,

//розраховані параметри, тощо.

using System;

using System.Drawing;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

Console.ForegroundColor = ConsoleColor.Cyan;

Console.Write("Введіть розмір матриці: ");

int S = Convert.ToInt32(Console.ReadLine());

Console.WriteLine($"Розмір: {S}");

int[][] Ar = new int[S][];

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

{

Ar[i] = new int[S];

}

int k = 1;

for (int start = 0; start < S; start++)

{

int row = start;

int col = 0;

while (row >= 0)

{

Ar[row][col] = k;

k++;

row--;

col++;

}

}

for (int start = 1; start < S; start++)

{

int row = S - 1;

int col = start;

while (col < S)

{

Ar[row][col] = k;

k++;

row--;

col++;

}

}

Console.WriteLine("Результат:");

for (int l = 0; l < S; l++)

{

for (int r = 0; r < S; r++)

{

if ( Ar[l][r] % 2 != 0)

{ Console.ForegroundColor = ConsoleColor.Red; }

else { Console.ForegroundColor = ConsoleColor.Blue;}

Console.Write($"{Ar[l][r],4} ");

}

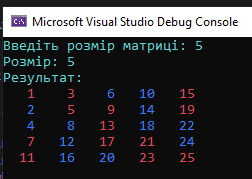
Console.WriteLine();

Console.ResetColor();

}

}

}



Завдання 9 Виконати підсумкове завдання згідно із номером свого варіанту

Визначити номера рядків і стовпців всіх сідлових точок матриці. Матриця А має сідлову точку, якщо A[i][j] є мінімальним елементом в і-у рядку і максимальним в j-у стовпці.

//Визначити номера рядків і стовпців всіх сідлових точок матриці.

//Матриця А має сідлову точку, якщо A[i][j] є мінімальним елементом в

//і-у рядку і максимальним в j-у стовпці.

using System;

using System.Text;

class Program

{

static void Main()

{

Console.OutputEncoding = UTF8Encoding.UTF8;

Console.ForegroundColor = ConsoleColor.Cyan;

Random rand = new Random();

bool foundSaddlePoint = false;

while (!foundSaddlePoint)

{

int S = 5;

int[,] Ar = new int[S, S];

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

{

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

{

Ar[i, j] = rand.Next(1, 100);

}

}

Console.WriteLine("Матриця:");

for (int l = 0; l < S; l++)

{

for (int r = 0; r < S; r++)

{

if (Ar[l, r] % 2 != 0)

{

Console.ForegroundColor = ConsoleColor.Red;

}

else

{

Console.ForegroundColor = ConsoleColor.Blue;

}

Console.Write($"{Ar[l, r],4} ");

}

Console.WriteLine();

Console.ResetColor();

}

// Пошук

foundSaddlePoint = false;

Console.WriteLine("Сідлові точки:");

// Пошук мінімального значення с рядку

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

{

int minRowValue = Ar[i, 0];

int minColIndex = 0;

for (int j = 1; j < S; j++)

{

if (Ar[i, j] < minRowValue)

{

minRowValue = Ar[i, j];

minColIndex = j;

}

}

// Перевірка, чи є мінімальний елемент у рядку максимальним у своєму стовпці

bool isSaddlePoint = true;

for (int row = 0; row < S; row++)

{

if (Ar[row, minColIndex] > minRowValue)

{

isSaddlePoint = false;

break;

}

}

if (isSaddlePoint)

{

Console.WriteLine($"Сідлова точка на позиції ({i}, {minColIndex}) з значенням {minRowValue}");

foundSaddlePoint = true; // Знайдено

}

}

}

Console.WriteLine("Матриця з сідловою точкою знайдена.");

Console.WriteLine("Пояснення: Матриця А має сідлову точку, якщо A[i][j] є мінімальним елементом в і-у рядку і максимальним в j-у стовпці.");

}

}

