**Алгоритми та структури даних. Основи алгоритмізації**

*Додаток 1*

Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний  інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра автоматизованих систем обробки інформації

і управління

Звіт

з лабораторної роботи № 9 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження лінійних алгоритмів»

Варіант\_\_\_21\_\_

Виконав студент \_\_\_Мартаков Данило Кирилович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 (шифр, прізвище, ім'я, по батькові)

Перевірив \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 ( прізвище, ім'я, по батькові) 

Київ 2021

**Назва роботи:** Дослідження Алгоритмів Обходу Масивів  
**Мета:** дослідити алгоритми обходу масивів, набути практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

**Варіант**: 21  
**Умова задачі**: 

**Постановка задачі:**

Потрібно створити двовимірний масив. У кожному рядку за допомогою алгоритму знайти максимальний елемент і його індекс за допомогою вкладених циклів. Після знаходження мінімальних елементів обміняти їх з першим елементом рядка.

**Побудова математичної моделі:** складемо таблицю імен змінних.

|  |  |  |  |
| --- | --- | --- | --- |
| **Змінна** | **Тип** | **Ім’я** | **Призначення** |
| матриця | Двох вимірний масив чисел | A[,] | Двохвимірний масив |
| Висота масива | Ціле Число | m | Початкове дане |
| Ширинамасива | Ціле Число | n | Початкове дане |
| Рандом | Random | rnd | Допоміжне значення |
| Значення Макс елемента в рядку | Ціле число | localMax | Допоміжне значення |
| Місцезнаходження Макс елемента в рядку по вертикалі | Ціле число | localMaxY | Допоміжне значення |
| Місцезнаходження Макс елемента в рядку по горизонталі | Ціле число | localMaxX | Допоміжне значення |
| Тимчасова змінна | Ціле число | temp | Допоміжне значення |
| Функція заповненя матриці випадковими числами | Функція | FillMatrix(matrix[,], row, colum) | Допоміжна Функція |
| Функція виводу значень матриці на екран | Функція | MatrixOutput(matrix[,], row, colum) | Допоміжна Функіця |
| Функція знаходження та заміни максимальних значень першого та останнього рядків | Функція | FindAndSwap(matrix[,], row, colum) | Допоміжна Функіця |

**Крок 1: Визначимо основні дії  
Крок 2: Деталізуємо основні зміні та визови функцій  
Крок 3: Деталізуємо тіло функції FillMatrix – заповнення матриці випадковими числами  
Крок 4: Деталізуємо тіло функції MatrixOutput – вивід значень матриці на екран  
Крок 5: Деталізуємо функцію FindAndSwap – фінкція знаходження першого та останнього максимальних елементів у матриці та їх обмін друг між другом**

**Псевдокод**

Крок 1

Початок

Введеня значень m та n

створення матриці A

заповнення матриці

вивід матриці

пошук та заміна максимальних елементів

вивід матриці

Кінець

Крок 2

Початок

m, n

A[m,n]

FillMatrix(A, m, n)

MatrixOutput(A, m, n)

FindAndSwap(A, m, n)

MatrixOutput(A, m, n)

Кінець

Крок 3 (псевдокод функції FillMatrix(matrix[,], int row, int colum))

Початок

rnd

для i = 0; i < row; i++

для j = 0; j < colum; j++

matrix[i,j] = rnd.Next(-50, 50)

все для

все для

Кінець

Крок 4 (псевдокод функції MatrixOutput(matrix[,], int row, int colum))

Початок

для i = 0; i < row; i++

для j = 0; j < colum; j++

Вивести matrix[i, j] + "\t"

все для

Вивести пусту строку

все для

Кінець

Крок 5 (псевдокод функції FindAndSwap(matrix[,], int row, int colum))

для i = 0; i < row; i++

localMax = -51

localMaxY = 0

localMaxX = 0

для j = 0; j < colum; j++

якщо matrix[i, j] > localMax

localMax = matrix[i, j]

localMaxY = i

localMaxX = j

все якщо

все для

temp = localMax

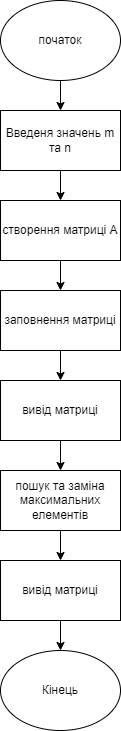
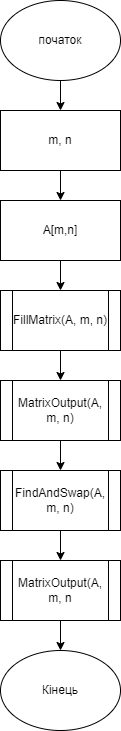
matrix[localMaxY, localMaxX] = matrix[firstMaxY, firstMaxX]

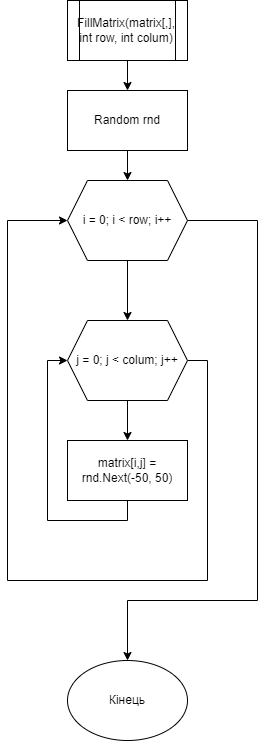
matrix[firstMaxY, firstMaxX] = temp

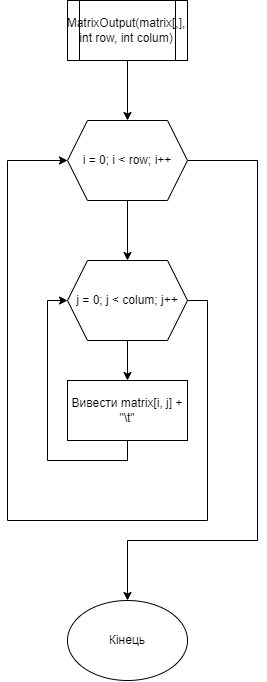
все для

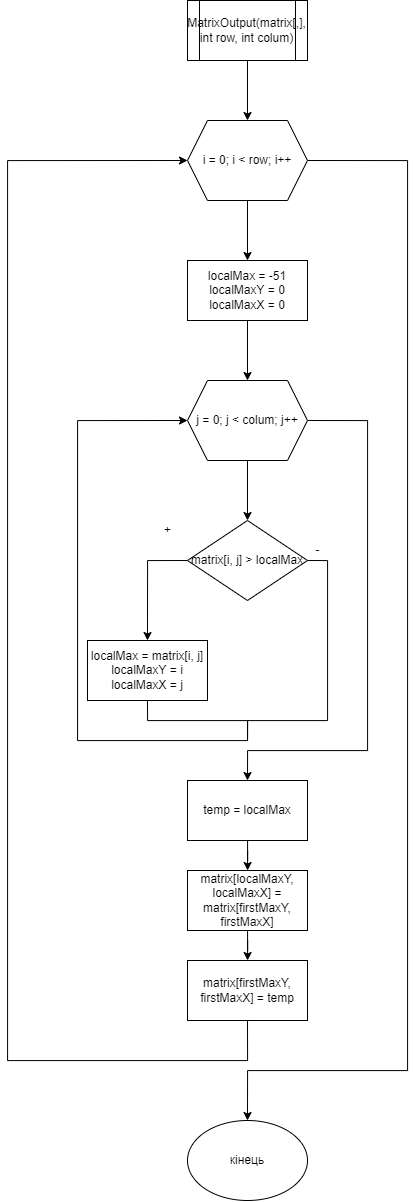
Кінець

**Блок Схема**







**Код Алгоритму**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace asdlab9

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter m");

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

Console.WriteLine("Enter n");

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

int[,] A = new int[m, n];

FillMatrix(A, m, n);

MatrixOutput(A, m, n);

FindAndSwap(A, m, n);

MatrixOutput(A, m, n);

Console.ReadLine();

}

public static void FillMatrix(int[,] matrix, int row, int colum)

{

Random rnd = new Random();

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

{

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

{

matrix[i, j] = rnd.Next(-50, 50);

}

}

}

public static void MatrixOutput(int[,] matrix, int row, int colum)

{

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

{

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

{

Console.Write(matrix[i, j] + "\t");

}

Console.WriteLine();

}

Console.WriteLine();

}

public static void FindAndSwap(int[,] matrix, int row, int colum)

{

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

{

int localMax = -51;

int localMaxY = 0;

int localMaxX = 0;

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

{

if (matrix[i, j] > localMax)

{

localMax = matrix[i, j];

localMaxY = i;

localMaxX = j;

}

}

Console.WriteLine("Max Element in row" + (i+1) + " = " + matrix[localMaxY, localMaxX]);

int temp = localMax;

matrix[localMaxY, localMaxX] = matrix[i, 0];

matrix[i, 0] = temp;

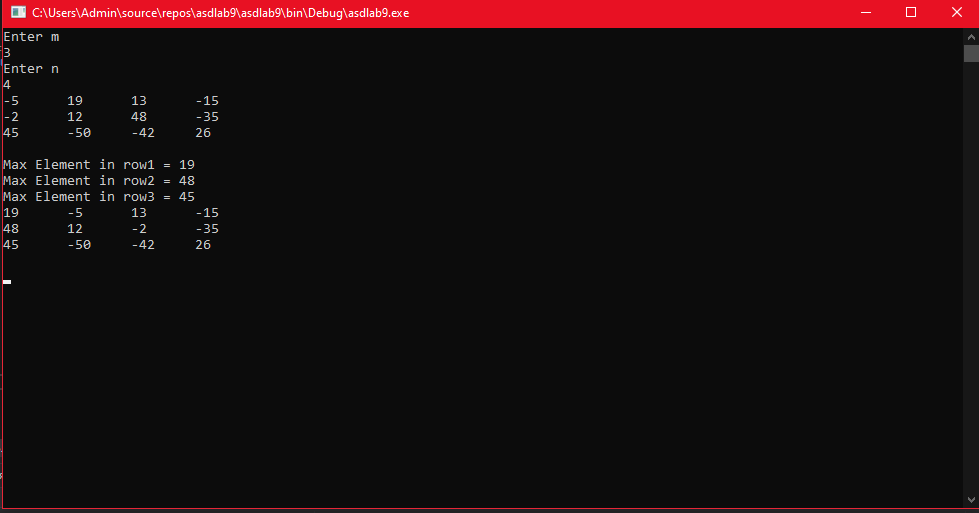
}

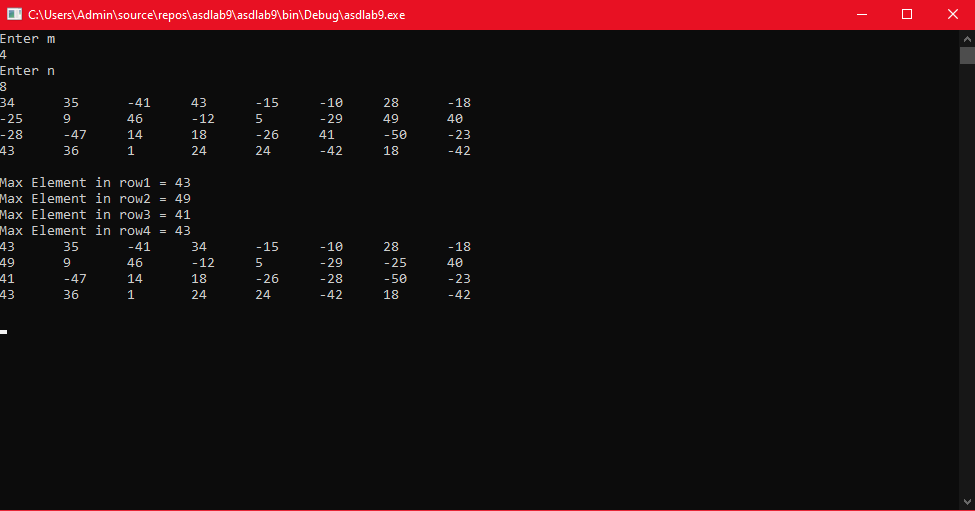
}

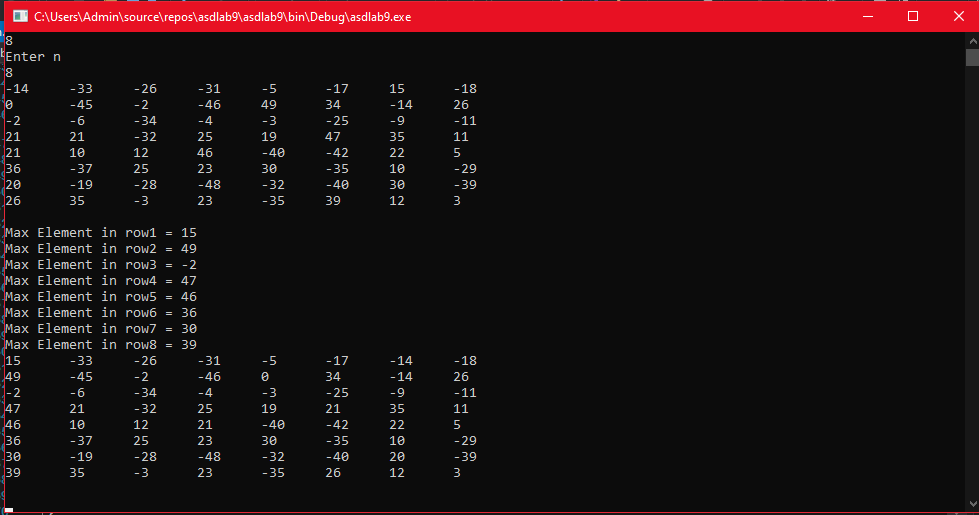
}

}

**Випробування алгоритму**

****

****

****

**Висновок**На прикладі даного завдання було досліджено алгоритми обходу масивів, набуто практичних навичок використання цих алгоритмів під час складання програмних специфікацій. Було створено псевдокод програми та власне код програми на мові с#