

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут імені
Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 9 з дисципліни «Алгоритми та
структури даних-1. Основи алгоритмізації»

«Дослідження лінійних алгоритмів»

Варіант 25

Виконав студент

Павленко Микита Андрійович

(шифр, прізвище, ім'я, по батькові)

Перевірив

Вечерковська Анастасія Сергіївна

(прізвище, ім'я, по батькові)

Лабораторна робота 9 ДСЛІДЖЕННЯ АЛГОРИТМІВ ОБХОДУ МАСИВІВ

Мета – дослідити алгоритми обходу масивів, набути практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

Варіант 25

Завдання

Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опису змінної індексованого типу (двовимірний масив) згідно з варіантом (табл. 1).
2. Ініціювання змінної, що описана в п.1 даного завдання.
3. Обчислення змінної, що описана в п.1, згідно з варіантом.

Постановка задачі

25 Задано матрицю дійсних чисел $A[m,n]$. У кожному рядку матриці знайти останній мінімальний елемент X і його місцезнаходження. Обміняти знайдене значення X з елементом середнього стовбця.

1) Побудова математичної моделі

Таблиця імен змінних:

Змінна	Тип	Ім'я	Призначення
Масив	Двовимірний масив дійсних чисел	array	Вхідні дані
Розмірність масиву	Цілий	m, n	Вхідні дані
Ітератори	Цілий	i, j	Проміжні дані
Тимчасова змінна	Дійсний	tmp	Проміжні дані
Мінімум рядка масиву	Дійсний	min	Проміжні дані
Позиція мінімального елемента	Цілий	min_j	Проміжні дані

Отже, математичне формулювання задачі зводиться до наповнення масиву розміром $m \times n$ (де n – непарне число, адже за умовою потрібно обмінювати елемент середнього стовбця) дійсними значеннями за допомогою `Random`. Надалі виконується пошук мінімального елемента рядка i і обмін його місцезнаходження з елементом середнього стовбця цього ж рядка.

2) Псевдокод алгоритму

Крок 1:

Функція Fill(double [,] array)

Для **i** від 0 до **m** повторити

Для **j** від 0 до **n** повторити

array[i, j] = rand.Next(-50, 50) + rand.NextDouble()

все повторити

все повторити

Все функція

початок

Ініціалізація змінних **tmp, min, min_j, m, n, array**

Виклик функції **Fill**

Для **i** від 0 до **m** повторити

min = array[i, 0]

Для **j** від 0 до **n** повторити

Якщо **min >= array[i, j]** то

min = array[i, j]

min_j = j

все повторити

tmp = array[i, n / 2]

array[i, n / 2] = min

array[i, min_j] = tmp

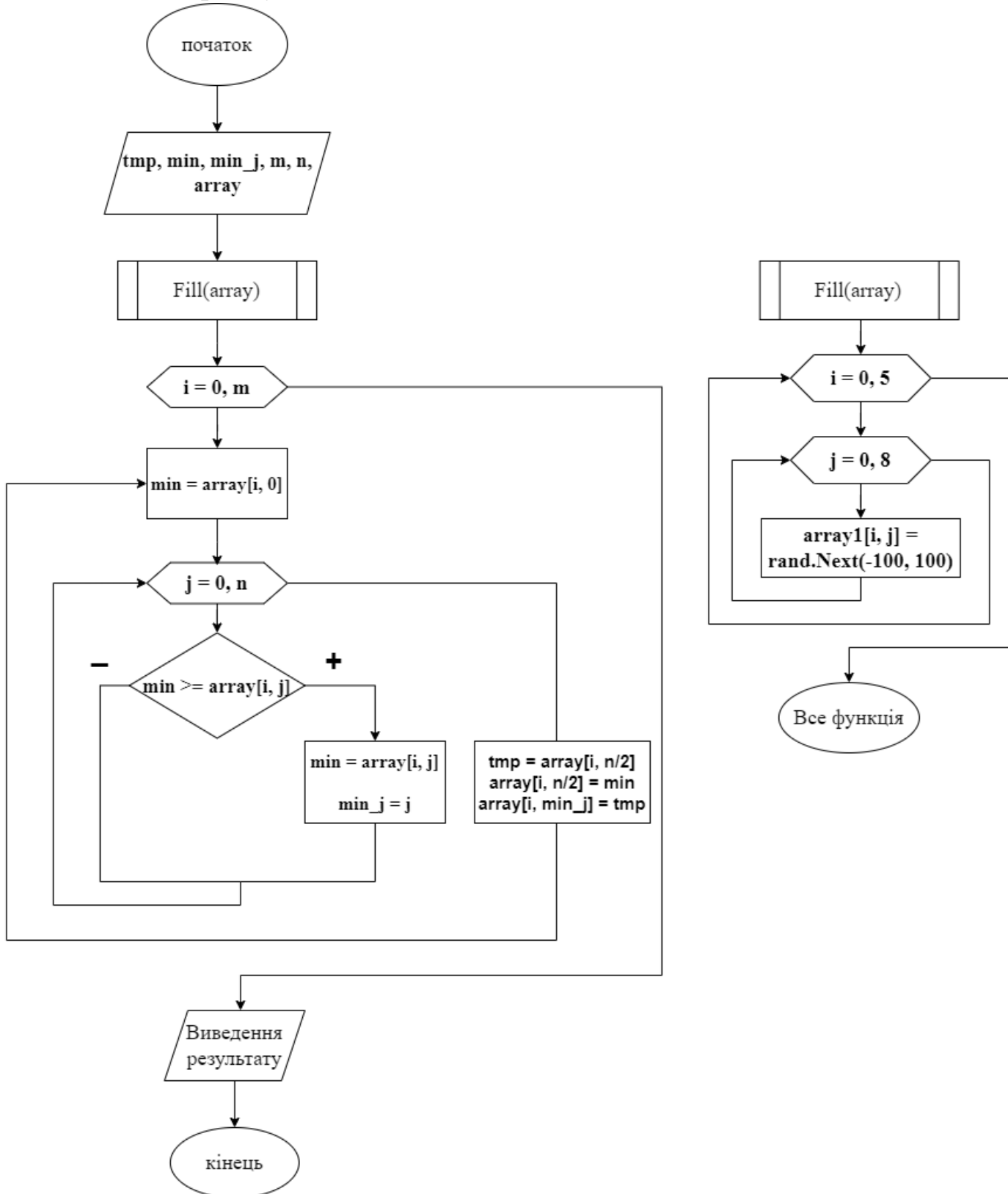
все повторити

Виведення результату

кінець

Алгоритми та структури даних. Основи алгоритмізації

3) Блок-схема алгоритму



5) Код програми

```
5  class Program
6  {
7      Console.WriteLine("Task 1: Fill array with random numbers");
8      static void Fill(double[,] array)
9      {
10         Random rand = new Random();
11         for (int i = 0; i < array.GetLength(0); i++)
12         {
13             for (int j = 0; j < array.GetLength(1); j++)
14             {
15                 array[i, j] = Math.Round((rand.Next(-50, 50) + rand.NextDouble()), 1);
16             }
17         }
18     }
19
20     Console.WriteLine("Task 2: Print array");
21     static void PrintArray(double[,] array)
22     {
23         for (int i = 0; i < array.GetLength(0); i++)
24         {
25             for (int j = 0; j < array.GetLength(1); j++)
26             {
27                 Console.Write(array[i, j] + " ");
28             }
29             Console.WriteLine();
30         }
31     }
32
33     Console.WriteLine("Task 3: Find min and max in array");
34     static void Main(string[] args)
35     {
36         double max;
37         double min;
38         int min_j = 0;
39         Console.WriteLine("Enter matrix size. m: ");
40         int m = Convert.ToInt32(Console.ReadLine());
41         Console.WriteLine("Enter matrix size. n: ");
42         int n = Convert.ToInt32(Console.ReadLine());
43         double[,] array = new double[m,n];
44         Fill(array);
45         PrintArray(array);
46         FindMinMax(array, out max, out min, out min_j);
47         Console.WriteLine($"Max: {max}, Min: {min}, Min index: {min_j}");
48     }
49 }
```

```
41  
42     Fill(array);  
43     PrintArray(array);  
44     Console.WriteLine();  
45  
46     for (int i = 0; i < m; i++)  
47     {  
48         min = array[i, 0];  
49         for (int j = 0; j < n; j++)  
50         {  
51             if (min >= array[i, j])  
52             {  
53                 min = array[i, j];  
54                 min_j = j;  
55             }  
56         }  
57         tmp = array[i, n / 2];  
58         array[i, n / 2] = min;  
59         array[i, min_j] = tmp;  
60     }  
61  
62     PrintArray(array);  
63 }  
64 }  
65 }
```

Результат роботи програми

1)

```
Консоль отладки Microsoft Visual Studio  
Enter matrix size m: 5  
Enter matrix size n: 7  
-27,3 -26,4 37,2 -32,9 -28,4 -10,1 -4,3  
3,7 -4 -38,1 -8,2 -8,4 -45,3 -37,6  
-34,1 30,8 -38,9 11,4 39,6 13,6 -48,8  
-18,2 -23,7 -22,7 -13,3 4,6 -36,8 5,4  
-25,4 6,2 -41,2 45,3 28,6 30,3 -43,9  
  
-27,3 -26,4 37,2 -32,9 -28,4 -10,1 -4,3  
3,7 -4 -38,1 -45,3 -8,4 -8,2 -37,6  
-34,1 30,8 -38,9 -48,8 39,6 13,6 11,4  
-18,2 -23,7 -22,7 -36,8 4,6 -13,3 5,4  
-25,4 6,2 -41,2 -43,9 28,6 30,3 45,3
```

2)

 Консоль отладки Microsoft Visual Studio

```
Enter matrix size m: 3
Enter matrix size n: 5
49,7 12,5 -1,8 26,6 40,4
-14,1 -3,7 2,1 34,9 -12
18 23,5 20,1 35,3 9,4

49,7 12,5 -1,8 26,6 40,4
2,1 -3,7 -14,1 34,9 -12
18 23,5 9,4 35,3 20,1
```

6) Висновки

Я дослідив алгоритми обходу масивів, набув практичних навичок використання цих алгоритмів під час складання програмних специфікацій.