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 алгоритмів обходу масивів»

Варіант 22

Виконав студент ІП-12 Мельник Михайло Олександрович

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

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 9**

**Дослідження алгоритмів обходу масивів**

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

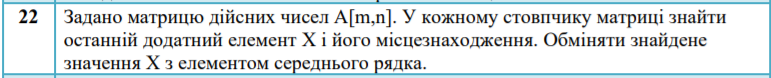
**Варіант 22**

**Задача:**

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 (табл. 1).

**Розв’язання:**

1. **Постановка задачі**

Результатом розв’язку задачі є оброблена матриця дійсних чисел, а також значення та позиція останнього додатного елемента X у кожному стовпчикові. Для його визначення не потрібні початкові значення.

1. **Побудова математичної моделі**

***Складемо таблицю імен змінних***

|  |  |  |  |
| --- | --- | --- | --- |
| **Змінна** | **Тип** | **Ім’я** | **Призначення** |
| Матриця | Індексований дійсний | A | Результат |
| Діапазон значень елементів масиву | Цілий | RANGE | Початкове дане |
| Кількість рядків | Цілий | m | Початкове дане |
| Кількість стовпців | Цілий | n | Початкове дане |
| Лічильник i | Порядковий | i | Проміжне дане |
| Лічильник j | Порядковий | j | Проміжне дане |
| Останній додатний елемент стовпчика | Дійсний | x | Проміжне дане, результат |
| Індекс рядка останнього додатного елемента стовпчика | Цілий | xInd | Проміжне дане, результат |

Таким чином, математичне формулювання задачі зводиться до:

* 1. Опису констант.
  2. Опису та ініціювання матриці випадковими значеннями у проміжку від RANGE до -RANGE обходом за рядками та її вивід за допомогою функції displayArray.
  3. Знаходження останнього додатного елементу стовпчика x та рядка, в якому він знаходиться обходом матриці за стовпцями, виводу цих значень.
  4. Обміну x з елементом середнього рядка кожного стовпця, індекс рядка обчислюємо цілочисельним діленням m на 2
  5. Виводу матриці після обміну за допомогою функціх displayArray.

1. **Розроблення алгоритму**

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.

*Крок 1.* Визначимо основні дії.

*Крок 2.* Деталізуємо дію опису констант

*Крок 3.* Деталізуємо дію опису та ініціювання матриці, виводу значень її елементів

*Крок 4.* Деталізуємо дію знаходження останнього додатного елемента елемента стовпчика та його виводу.

*Крок 5.* Деталізуємо дію обміну елементів матриці.

*Крок 6.* Деталізуємо дію виводу значень елементів обробленої матриці.

***Псевдокод***

*крок 1*

**початок**

опис констант

опис та ініціювання матриці, вивід її елементів

знаходження останнього додатного

елемента стовпчика та його вивід

обмін елементів матриці

вивід значень елементів обробленої матриці

**кінець**

**displayArray(arr[m][n])**

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

**вивід** arr[i][j]

**все повторити**

**все повторити**

**кінець displayArray**

*крок 2*

**початок**

m := 5

n := 7

RANGE := 100

опис та ініціювання матриці, вивід її елементів

знаходження останнього додатного

елемента стовпчика та його вивід

обмін елементів матриці

вивід значень елементів обробленої матриці

**кінець**

**displayArray(arr[m][n])**

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

**вивід** arr[i][j]

**все повторити**

**все повторити**

**кінець displayArray**

*крок 3*

**початок**

m := 5

n := 7

RANGE := 100

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

A[i][j] = rand() / RAND\_MAX \* 2 \*

RANGE - RANGE

**все повторити**

**все повторити**

displayArray(A[m][n])

знаходження останнього додатного

елемента стовпчика та його вивід

обмін елементів матриці

вивід значень елементів обробленої матриці

**кінець**

**displayArray(arr[m][n])**

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

**вивід** arr[i][j]

**все повторити**

**все повторити**

**кінець displayArray**

*крок 4*

**початок**

m := 5

n := 7

RANGE := 100

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

A[i][j] = rand() / RAND\_MAX \* 2 \*

RANGE - RANGE

**все повторити**

**все повторити**

displayArray(A[m][n])

**повторити**

**для** j **від** 0 **до** n

x := 0

**повторити**

**для i** **від** 0 **до** m

**якщо** A[i][j] > 0

**то**

x := A[i][j]

xInd := i

**все якщо**

**все повторити**

**якщо** x != 0

**то**

**вивід** j, xInd, x

обмін елементів матриці

**інакше**

**вивід** “No positive numbers”, j

**все якщо**

**все повторити**

вивід значень елементів обробленої матриці

**кінець**

**displayArray(arr[m][n])**

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

**вивід** arr[i][j]

**все повторити**

**все повторити**

**кінець displayArray**

*крок 5*

**початок**

m := 5

n := 7

RANGE := 100

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

A[i][j] = rand() / RAND\_MAX \* 2 \*

RANGE - RANGE

**все повторити**

**все повторити**

displayArray(A[m][n])

**повторити**

**для** j **від** 0 **до** n

x := 0

**повторити**

**для i** **від** 0 **до** m

**якщо** A[i][j] > 0

**то**

x := A[i][j]

xInd := i

**все якщо**

**все повторити**

**якщо** x != 0

**то**

**вивід** j, xInd, x

A[xInd][j] = A[m div 2][j]

A[m div 2][j] = x

**інакше**

**вивід** “No positive numbers”, j

**все якщо**

**все повторити**

вивід значень елементів обробленої матриці

**кінець**

**displayArray(arr[m][n])**

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

**вивід** arr[i][j]

**все повторити**

**все повторити**

**кінець displayArray**

*крок 6*

**початок**

m := 5

n := 7

RANGE := 100

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

A[i][j] = rand() / RAND\_MAX \* 2 \*

RANGE - RANGE

**все повторити**

**все повторити**

displayArray(A[m][n])

**повторити**

**для** j **від** 0 **до** n

x := 0

**повторити**

**для i** **від** 0 **до** m

**якщо** A[i][j] > 0

**то**

x := A[i][j]

xInd := i

**все якщо**

**все повторити**

**якщо** x != 0

**то**

**вивід** j, xInd, x

A[xInd][j] = A[m div 2][j]

A[m div 2][j] = x

**інакше**

**вивід** “No positive numbers”, j

**все якщо**

**все повторити**

displayArray(A[m][n])

**кінець**

**displayArray(arr[m][n])**

**повторити**

**для** і **від** 0 **до** m

**повторити**

**для** j **від** 0 **до** n

**вивід** arr[i][j]

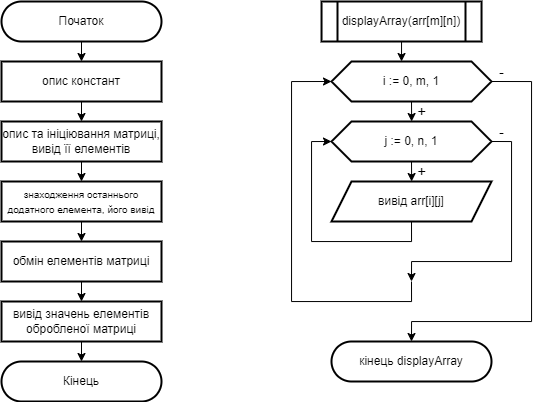
**все повторити**

**все повторити**

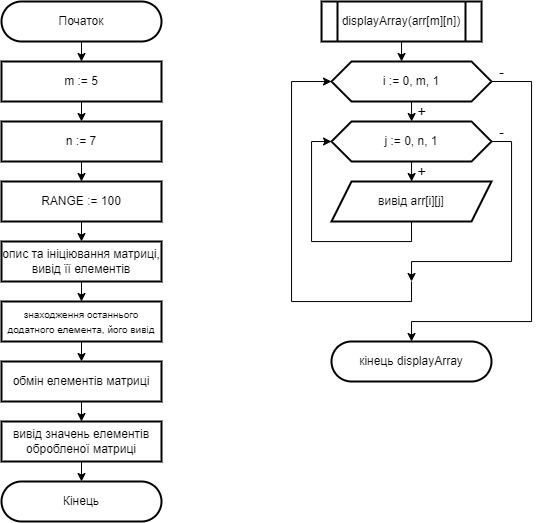
**кінець displayArray**

***Блок-схема***

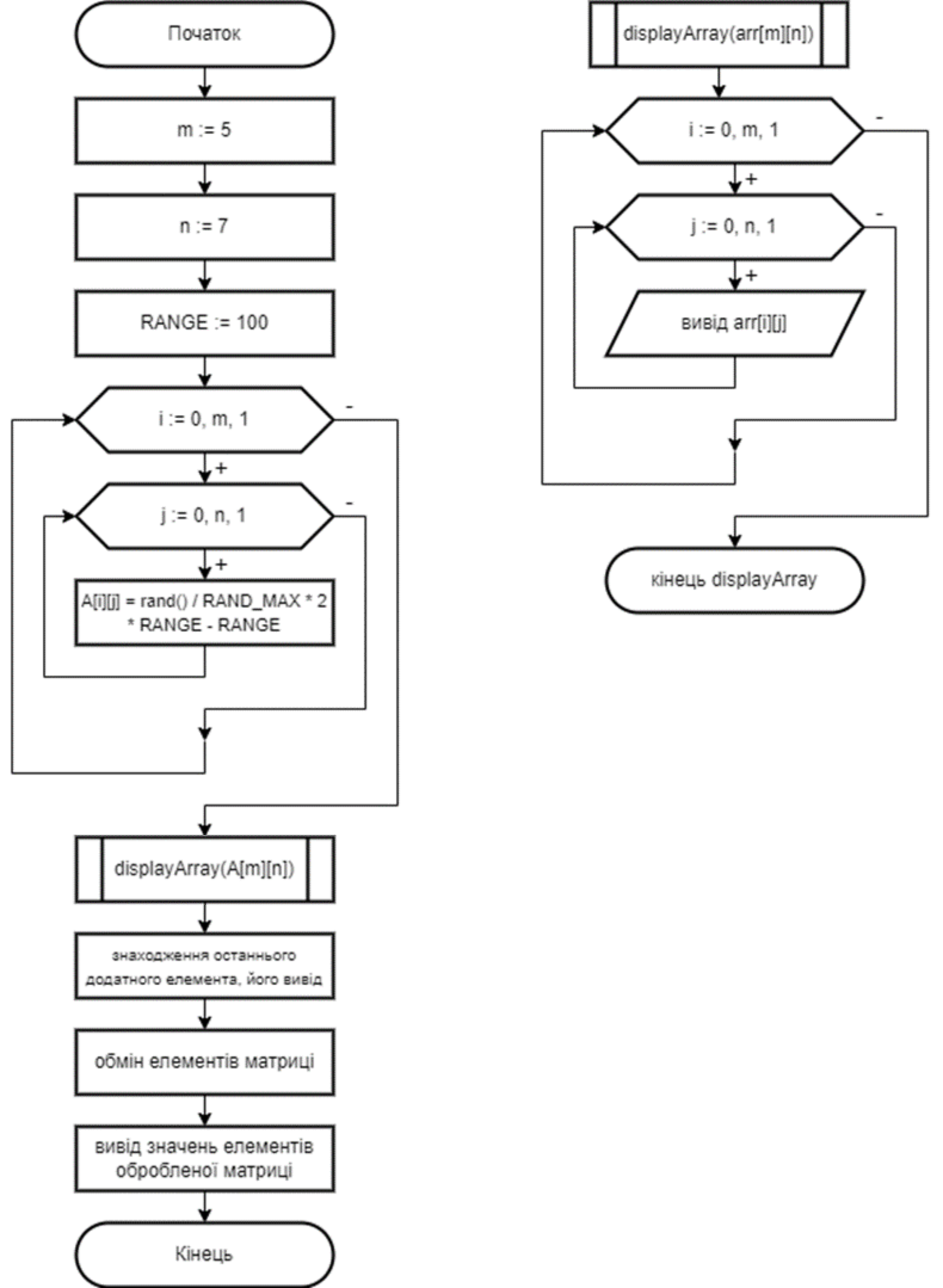
*Крок 1*

**

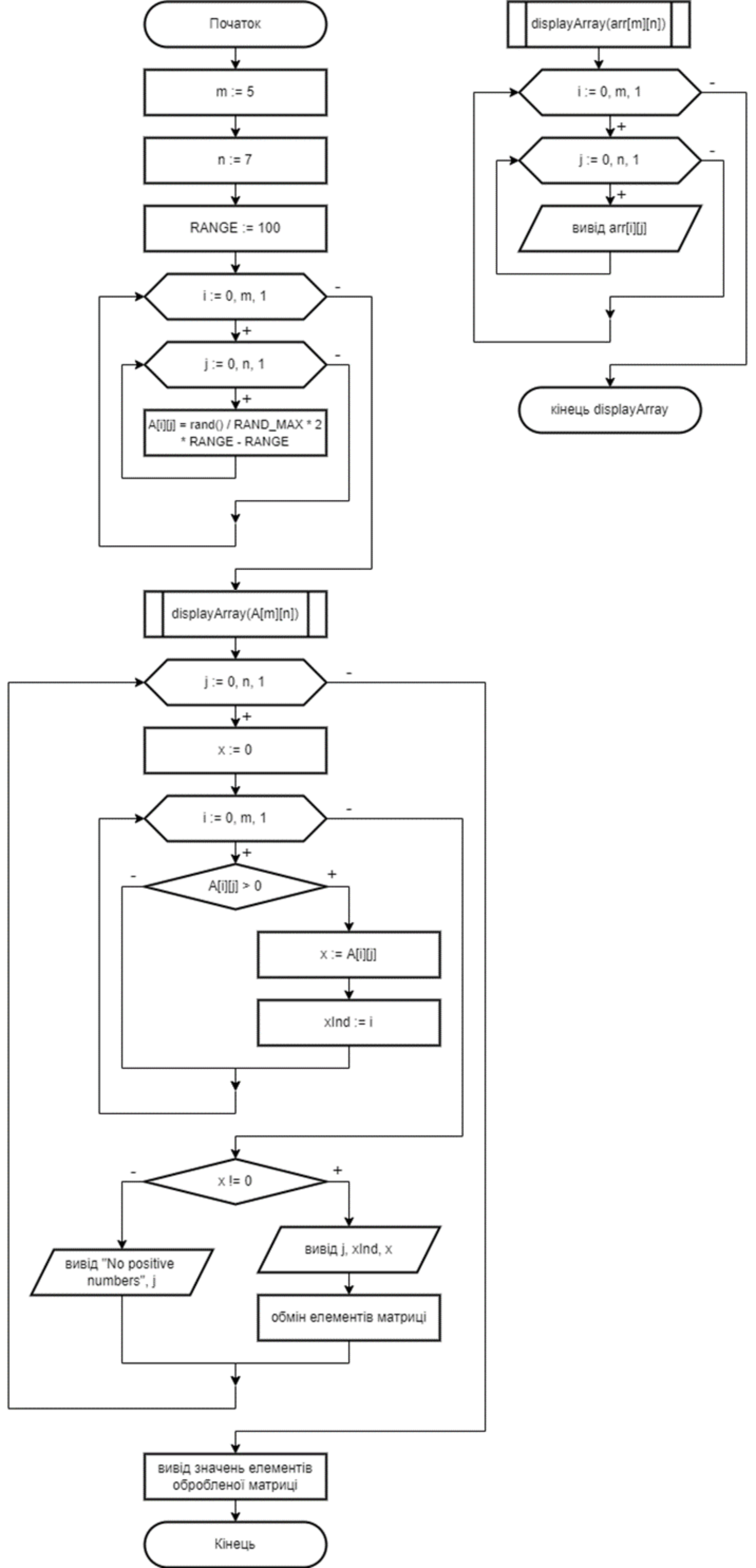
*Крок 2*

**

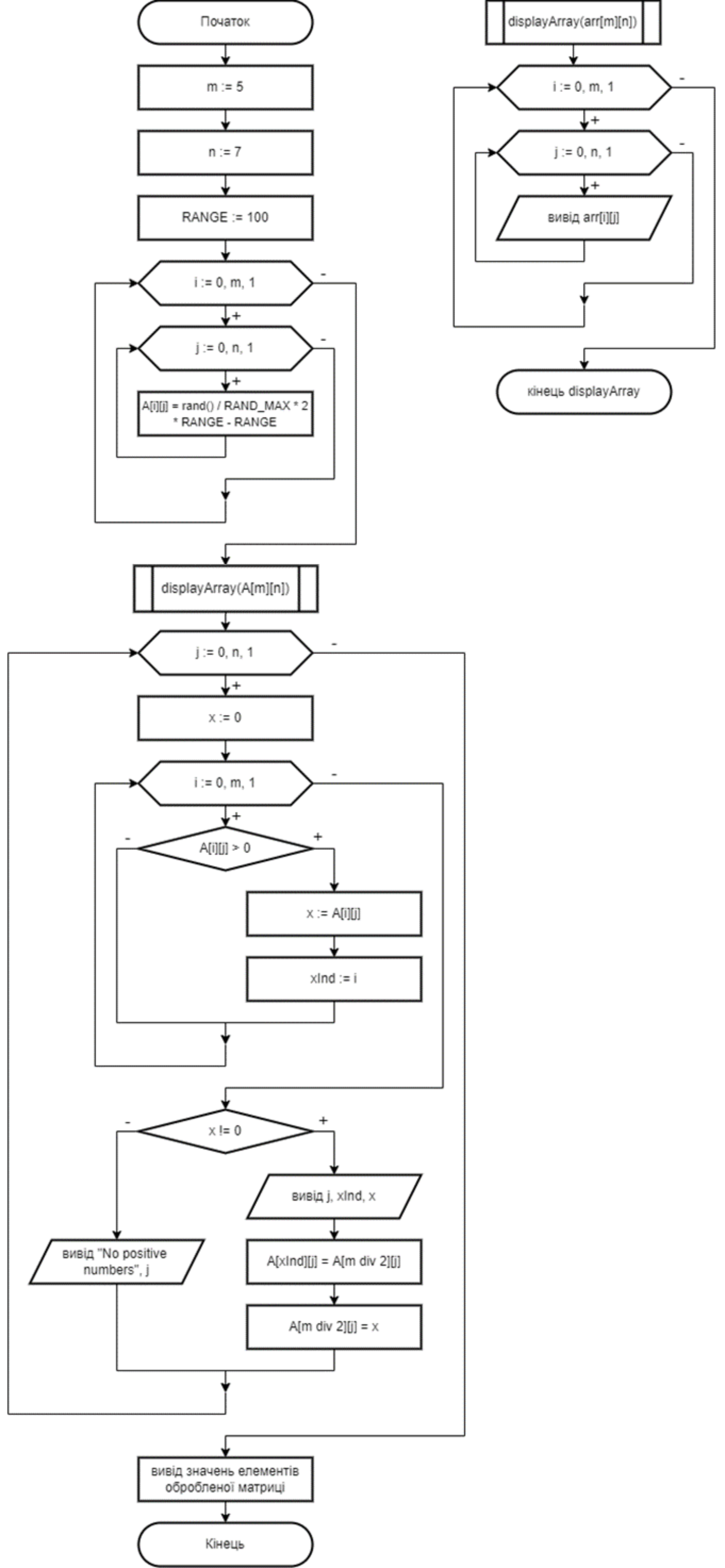
*Крок 3*

**

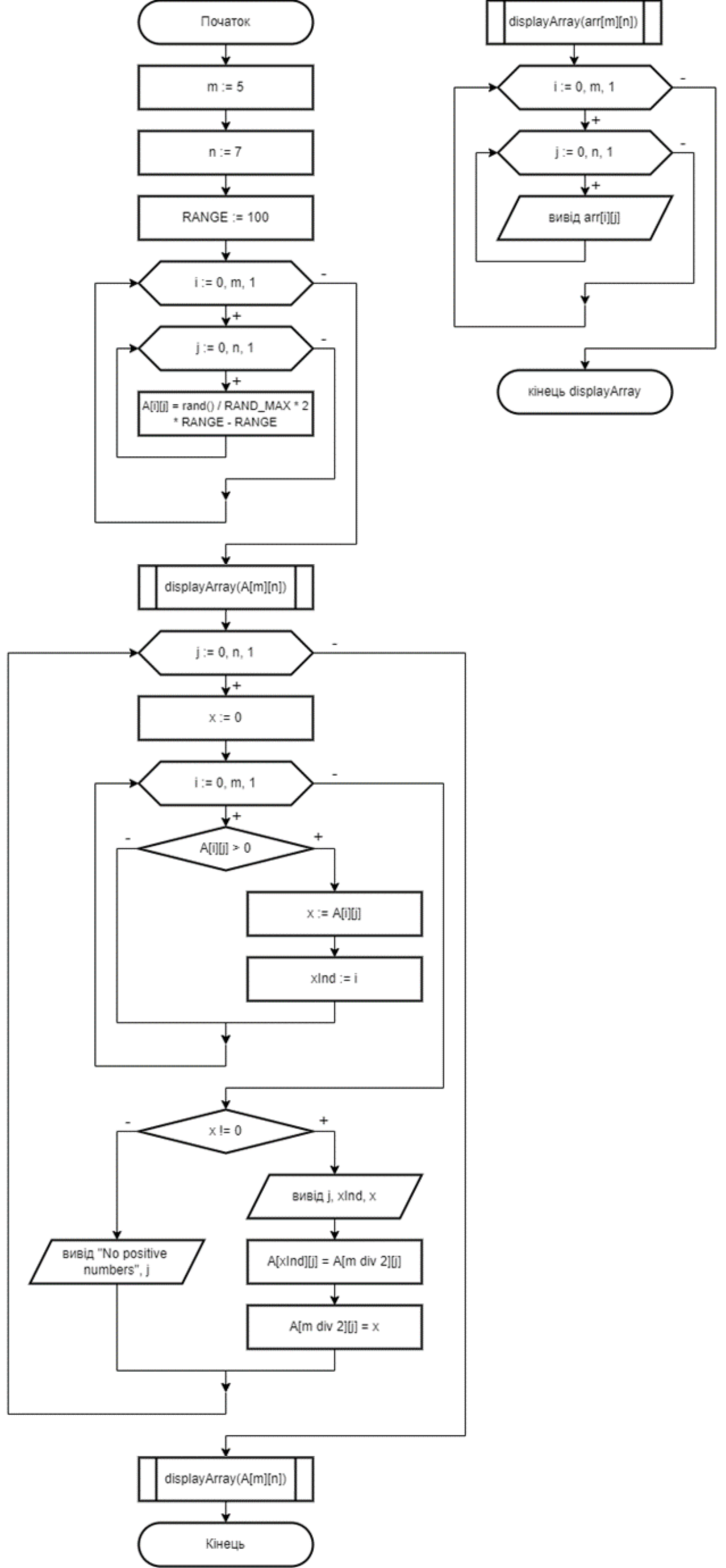
*Крок 4*

**

*Крок 5*

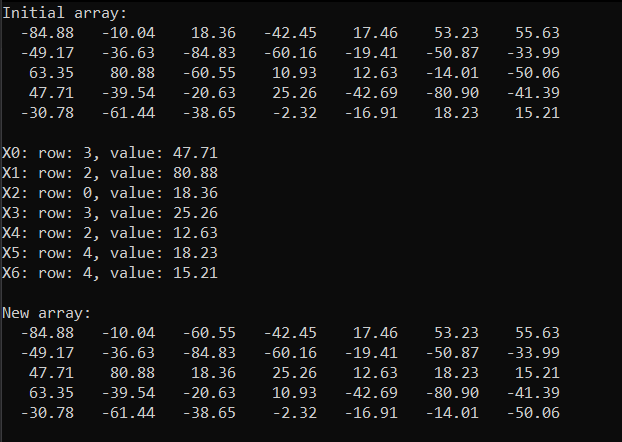
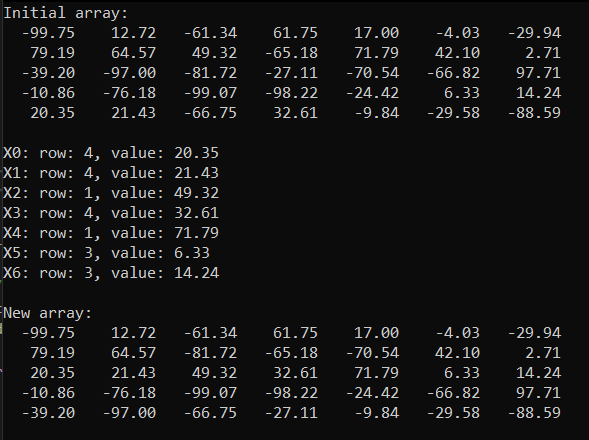
**

*Крок 6*

**

1. **Складання програми**
2. **Тестування програми**

Результат не залежить від вхідних даних, достатньо відеокопії результату



Програма відповідає поставленій задачі, помилок не виявлено.

1. **Висновки**

Під час виконання даної лабораторної роботи було розроблено й випробувано алгоритм обходу матриці за стовпцями, пошуку за стовпцями і обміну елементів матриці, досліджено алгоритми обходу масивів, набуто практичних навичок використання цих алгоритмів під час складання програмних специфікацій.