Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний

інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної инженериії

Звіт

з лабораторної работи № 9 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження алгоритмів обходу масивів»

Варіант 1

Виконав студент ІП-14 Аджигельдієва Мадіна Алімівна

Перевірів Мартинова Оксана Петрівна

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.

Задача 1. Розробити алгоритм та написати програму, яка складається з наступних дій:

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



**Розв’язання**

**Постановка задачі.** Результатом розв’язку є визначення присутності заданого дійсного числа Х та обмін знайденного значення Х з елементами побічної діагоналі.

**Побудова математичної моделі.**

|  |  |  |
| --- | --- | --- |
| **Змінна** | **Тип** | **Призначення** |
| M\_ROWS, N\_COLS | Цілий | Постійні змінні для визначення розміру масива |
| X | Дійсний | Постійна змінна для заданого числа |
| matrix2[M\_ROWS][N\_COLS] | Дійсний | Матриця чисел  (двовимірний масив) |
| i, j, size, k, temp | Цілий | Локальні змінні в основній функції |
| isXFound | Логічний |
| matrix2[i][j] | Дійсний | Результат  оновлена матриця |

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

Визначимо основні дії:

*Крок 1.* Введення const M\_ROWS, const N\_COLS, const X, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound, temp.

*Крок 2.* Ініціювання matrix2.

*Крок 3.* Знаходження найкоротшої сторони matrix2.

*Крок 4.* Обхід matrix2 за елементами побічної діагоналі, знаходження положення елемента Х та обмін елементів побічної діагоналі matrix2.

*Крок 5.* Ініціювання оновленої matrix2.

**Псевдокод алгоритму.**

*Крок 1.*

**початок**

Введення const M\_ROWS, const N\_COLS, const X, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound, temp

Ініціювання matrix2

Знаходження найкоротшої сторони matrix2

Обхід matrix2 за елементами побічної діагоналі, знаходження положення елемента Х та обмін елементів побічної діагоналі matrix2

Ініціювання оновленої matrix2

**кінець**

*Крок 2.*

**початок**

Введення const M\_ROWS = 4, const N\_COLS = 3, const X = 3, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound = true або false, temp

**для** (i = 0; i < M\_ROWS; i++)

**для** (j = 0; j < N\_COLS; j++)

**виведення** matrix2[i][j];

Знаходження найкоротшої сторони matrix2

Обхід matrix2 за елементами побічної діагоналі, знаходження положення елемента Х та обмін елементів побічної діагоналі matrix2

Ініціювання оновленої matrix2

**кінець**

*Крок 3.*

**початок**

Введення const M\_ROWS = 4, const N\_COLS = 3, const X = 3, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound = true або false, temp

**для** (i = 0; i < M\_ROWS; i++)

**для** (j = 0; j < N\_COLS; j++)

**виведення** matrix2[i][j];

**якщо** (M\_ROWS < N\_COLS)

M\_ROWS

**інакше** N\_COLS;

Обхід matrix2 за елементами побічної діагоналі, знаходження положення елемента Х та обмін елементів побічної діагоналі matrix2

Ініціювання оновленої matrix2

**кінець**

*Крок 4.*

**початок**

Введення const M\_ROWS = 4, const N\_COLS = 3, const X = 3, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound = true або false, temp

**для** (i = 0; i < M\_ROWS; i++)

**для** (j = 0; j < N\_COLS; j++)

**виведення** matrix2[i][j];

**якщо** (M\_ROWS < N\_COLS)

M\_ROWS

**інакше** N\_COLS;

**для** (i = 0; i < size; i++)

j = N\_COLS – 1 – i;

**для** (k = 0; k < N\_COLS; k++)

**якщо** (matrix2[i][j] == X)

isXFound = true;

break;

**якщо** (isXFound)

temp = matrix2[i][j];

matrix2[i][k] = matrix2[i][j];

matrix2[i][j] = temp;

Ініціювання оновленої matrix2

**кінець**

*Крок 5.*

**початок**

Введення const M\_ROWS = 4, const N\_COLS = 3, const X = 3, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound = true або false, temp

**для** (i = 0; i < M\_ROWS; i++)

**для** (j = 0; j < N\_COLS; j++)

**виведення** matrix2[i][j];

**якщо** (M\_ROWS < N\_COLS)

M\_ROWS

**інакше** N\_COLS;

**для** (i = 0; i < size; i++)

j = N\_COLS – 1 – i;

**для** (k = 0; k < N\_COLS; k++)

**якщо** (matrix2[i][j] == X)

isXFound = true;

break;

**якщо** (isXFound)

temp = matrix2[i][j];

matrix2[i][k] = matrix2[i][j];

matrix2[i][j] = temp;

**для** (i = 0; i < M\_ROWS; i++)

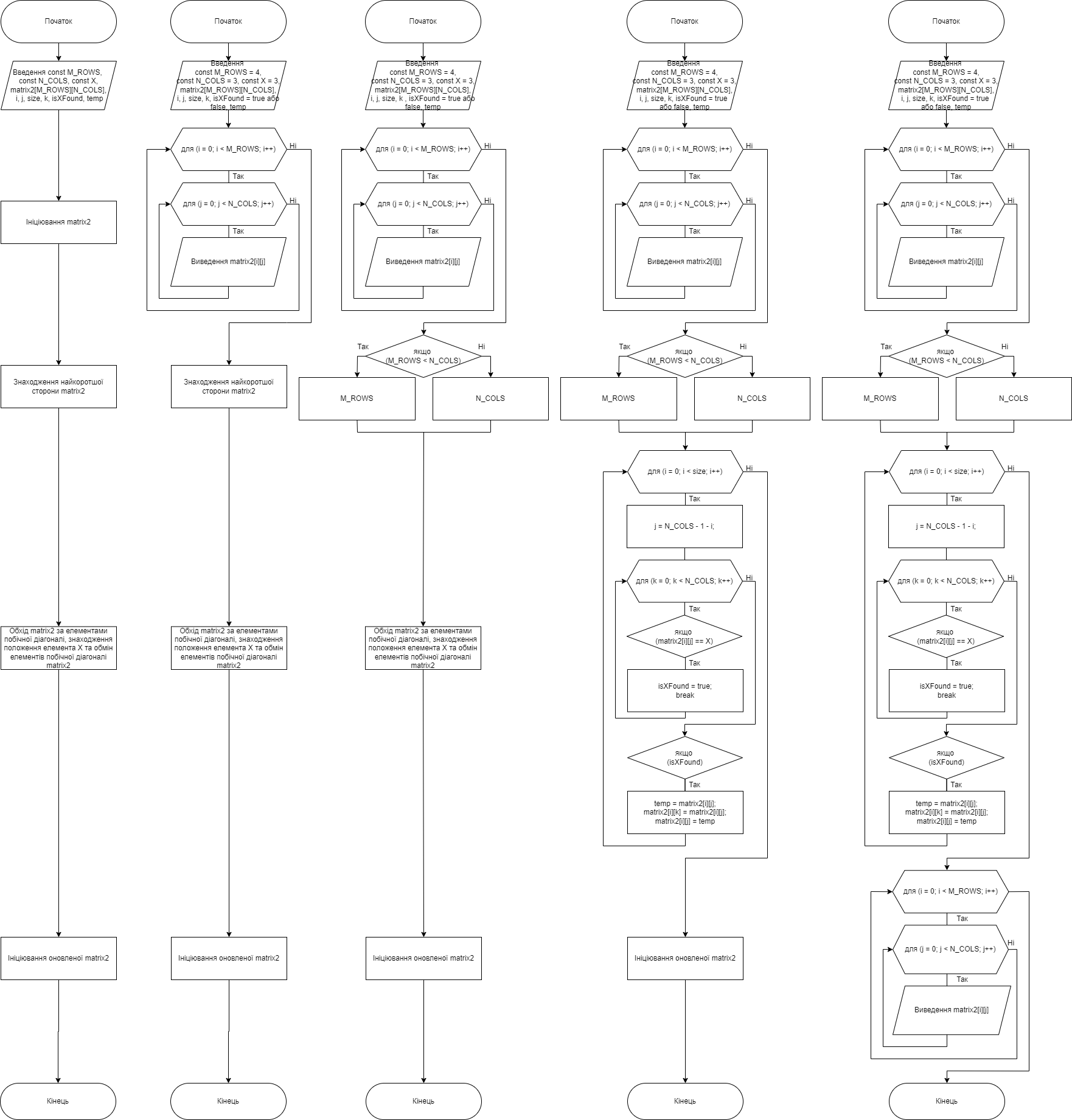
**для** (j = 0; j < N\_COLS; j++)

**виведення** matrix2[i][j]

**кінець**

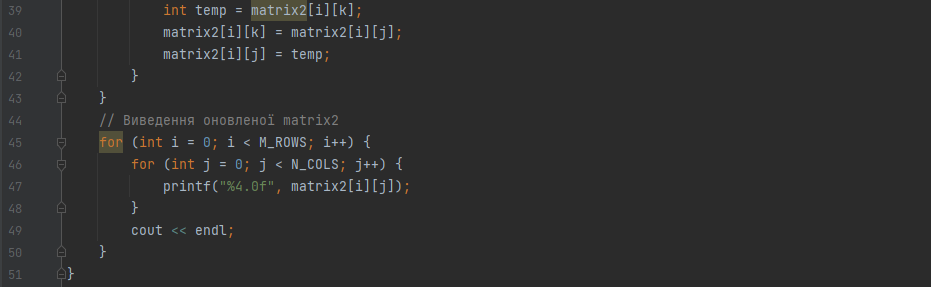
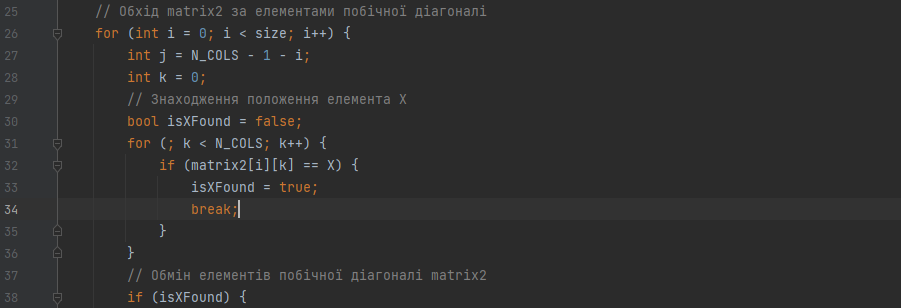
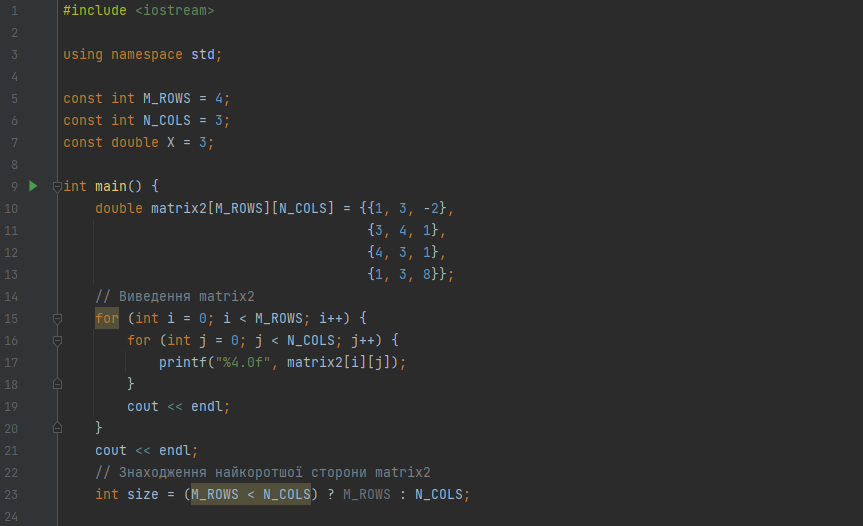
**Блок-схема алгоритму.**

*Крок 1 Крок 2 Крок 3 Крок 4 Крок 5*

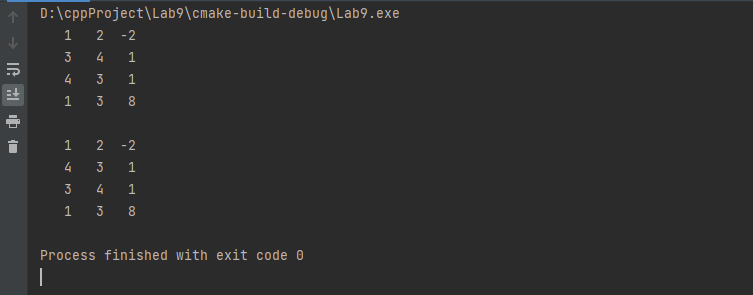


**Код програми.**

Представимо у вигляді коду програми та виведемо результат роботи.



**Результат коду програми.**



**Випробування алгоритму.**

Перевіримо правільність алгоритму на довільних конкретних значеннях початкових даних.

|  |  |
| --- | --- |
| **Блок** | **Дія** |
|  | **Початок** |
| 1 | **Введення** const M\_ROWS = 4, const N\_COLS = 3, const X = 3, matrix2[M\_ROWS][N\_COLS], i, j, size, k, isXFound = true або false, temp |
| 2 | **для** (i = 0; i < M\_ROWS; i++)  **для** (j = 0; j < N\_COLS; j++)  **виведення** matrix2[i][j];  1 3 -2  3 4 1  4 3 1  1 3 8 |
| 3 | **якщо** (M\_ROWS < N\_COLS)  M\_ROWS  **інакше** N\_COLS; |
| 4 | **для** (i = 0; i < size; i++)  j = N\_COLS – 1 – i;  **для** (k = 0; k < N\_COLS; k++)  **якщо** (matrix2[i][j] == X)  isXFound = true;  break;  **якщо** (isXFound)  temp = matrix2[i][j];  matrix2[i][k] = matrix2[i][j];  matrix2[i][j] = temp; |
| 5 | **для** (i = 0; i < M\_ROWS; i++)  **для** (j = 0; j < N\_COLS; j++)  **виведення** matrix2[i][j]  1 -2 3  4 3 1  3 4 1  1 3 8 |
|  | **Кінець** |

**Висновок.**

В результаті виконання лабораторної роботи:

- набула теоритичні знання о алгоритмах обходу масивів: порядок виконання алгоритмів обходу, види перебору рядків та стовбців.

- визначила основні кроки дій, записала програмні специфікації у псевдокоді, зобразила алгоритм у графічній формі у вигляді блок-схеми.

- представила програмні спеціфікації у вигляді коду програми на мові програмування С++, отримала правильний результат, виконала перевірку алгоритму підставляючи конкретні значення у певні дані.