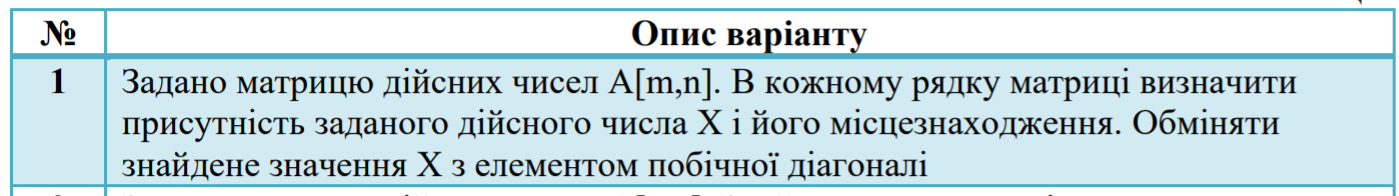
**Лабораторна робота 9**

**ДСЛІДЖЕННЯ АЛГОРИТМІВ ОБХОДУ МАСИВІВ**

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

**1)Постановка задачі**



Для коректного запуску програми треба ввести рівні між собою m та n, щоб побічна діагональ була чітко виражена.

**2)Побудова математичної моделі**

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім'я | Призначення |
| Вхідна змінна | Цілий | n | Кількість стовпців |
| Вхідна змінна | Цілий | m | Кількість рядків |
| Вхідна змінна | Цілий | x | Шукане число |
| Массив | Цілий | mas | Матриця чисел |
| Лічільники | Цілий | I,j | Лічільники |
| Функція | Цілий | input | Функція ініціалізування масиву |
| Функція | Цілий | swap | Функція переміщення элемента побічної діагоналі |

Крок 1 Опис двовимірного масиву

Крок 2 Ініціалізація функції заповнення двовимірного масиву

Крок 3 Ініціалізація функції обміну

**3)Псевдокод**

**Крок 1**

**Початок**

int\*\* mas = new int\*[m]

Для I від 0 до m з кроком 1

mas[i] = new int[n];

**Все повторити**

Ініціалізація функції заповнення двовимірного масиву

Ініціалізація функції обміну

**Кінець**

**Крок 2**

**Початок**

void input(int\*\* arr, int m, int n)

Початок

Для i Від 0 до m з кроком 1

Для j Від 0 до n з кроком 1

arr[i][j] = rand() % 15 + 1;

Все повторити

Все повторити

Кінець

int\*\* mas = new int\*[m]

Для I від 0 до m з кроком 1

as[i] = new int[n];

**Все повторити**

input(mas, m, n)

Ініціалізація функції обміну

**Кінець**

**Крок 3**

**Початок**

void swap(int\*\* mas, int m, int n,int x)

Початок

Для i Від 0 до m з кроком 1

Для j Від 0 до n з кроком 1

Якщо mas[i][j]== x

То mas[i][j]=mas[i][n-1-i]

Mas[i][n-1-i]=x

Все якщо

Все повторити

Все повторити

Кінець

int\*\* mas = new int\*[m]

Для I від 0 до m з кроком 1

mas[i] = new int[n];

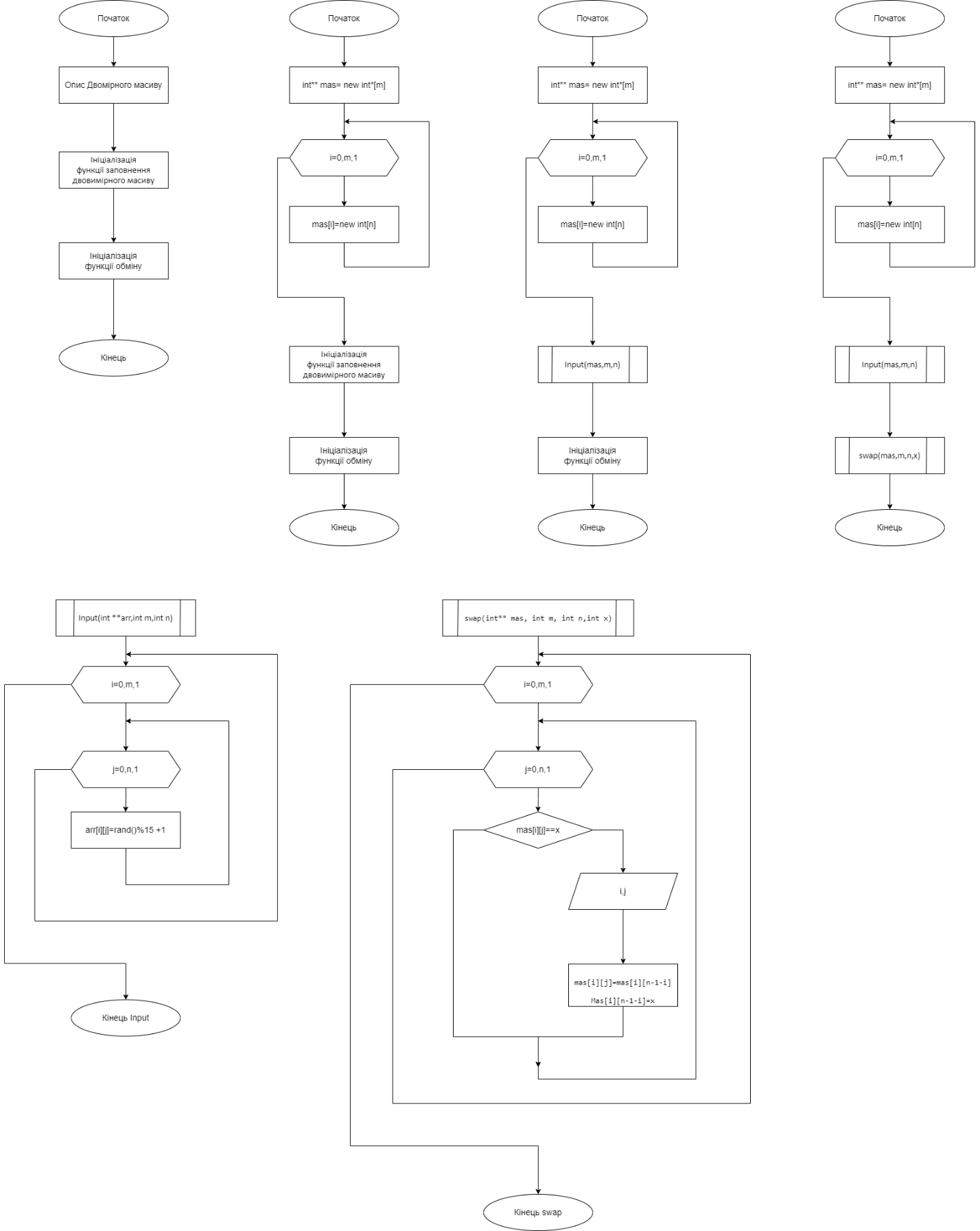
**Все повторити**

input(mas, m, n)

swap(mas, m, n, x)

**Кінець**

**4)Блок-схема**

****

**5)Код програми**

#include <iostream>

#include <iomanip>

#include <time.h>

using namespace std;

void input(int\*\* arr, int m, int n);

void output(int\*\* arr, int m, int n);

void swap(int\*\* mas, int m, int n, int x);

int main()

{

srand(time(NULL));

int m, n;

cout << "Input m, n - ";

cin >> m >> n;

int\*\* mas = new int\*[m];

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

mas[i] = new int[n];

input(mas, m, n);

output(mas,m,n);

int x;

cout << "Input X - ";

cin >> x;

swap(mas, m, n, x);

output(mas, m, n);

}

void input(int\*\* arr, int m, int n) {//Ввод массива

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

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

arr[i][j] = rand() % 15 + 1;

}

void output(int\*\* arr, int m, int n) {//Вывод массива

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

{

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

cout << setw(4) << arr[i][j];

cout << endl;

}

}

void swap(int\*\* mas, int m, int n,int x) {

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

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

if (mas[i][j] == x) {

cout << "[" << i << "," << j << "]" << endl;

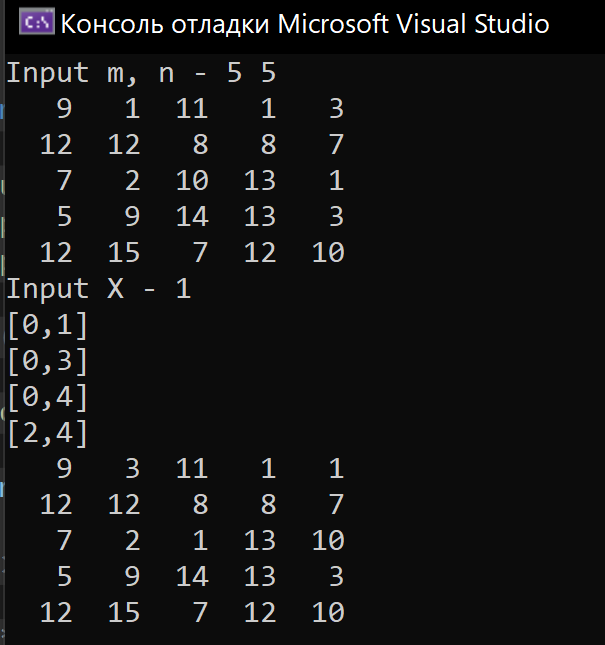
mas[i][j] = mas[i][n - 1 - i];

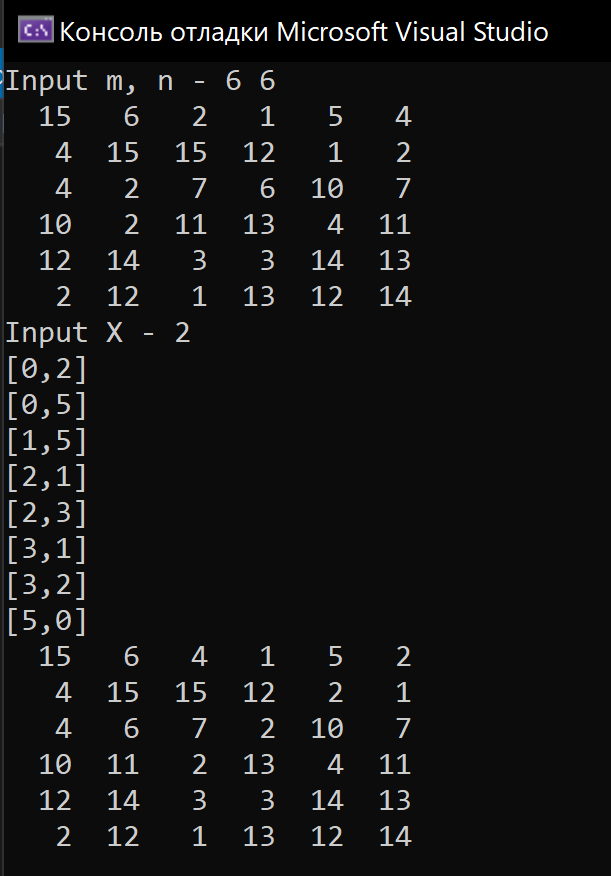
mas[i][n - 1 - i] = x;

}

}

**6)Тестування програми**





**7)Висновок**

У цій роботі було досліджено алгоритми обходу масивів, набуто практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

Специфікою даної задачі було переставлення елементів побічної діагоналі з даним другим елементом.