

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний аерокосмічний університет ім. М. Є. Жуковського  
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів  
Кафедра систем управління літальних апаратів

## **Лабораторна робота № 7**

з дисципліни «Алгоритмізація та програмування»  
на тему « " Реалізація алгоритмів обробки двовимірних масивів на  
мові C ++"»

ХАІ.301. Спеціальність: АВТОМАТИЗАЦІЯ, КОМП'ЮТЕРНО-  
ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА РОБОТОТЕХНІКА, група 319, номер у  
списку:28 ЛР

Виконав студент гр. 319  
Калашник О.С.

Перевірив  
\_\_\_\_\_ к.т.н., доц. Олена ГАВРИЛЕНКО

2023

## МЕТА РОБОТИ

Вивчити теоретичний матеріал з основ представлення двовимірних масивів

(матриць) на мові C ++ і реалізувати оголошення, введення з консолі, обробку і

виведення в консоль матриць на мові C ++ в середовищі Visual Studio.

## ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити завдання на аналіз і виведення елементів матриці.

Введення і виведення даних здійснити в командному вікні.

Завдання знаходиться у таблиці під номером Matrix20.

Завдання 2. Перетворити матрицю відповідно до свого варіанту завдання

Matrix53, розмір матриці і його елементи ввести з консолі. Вивести результати у консоль.

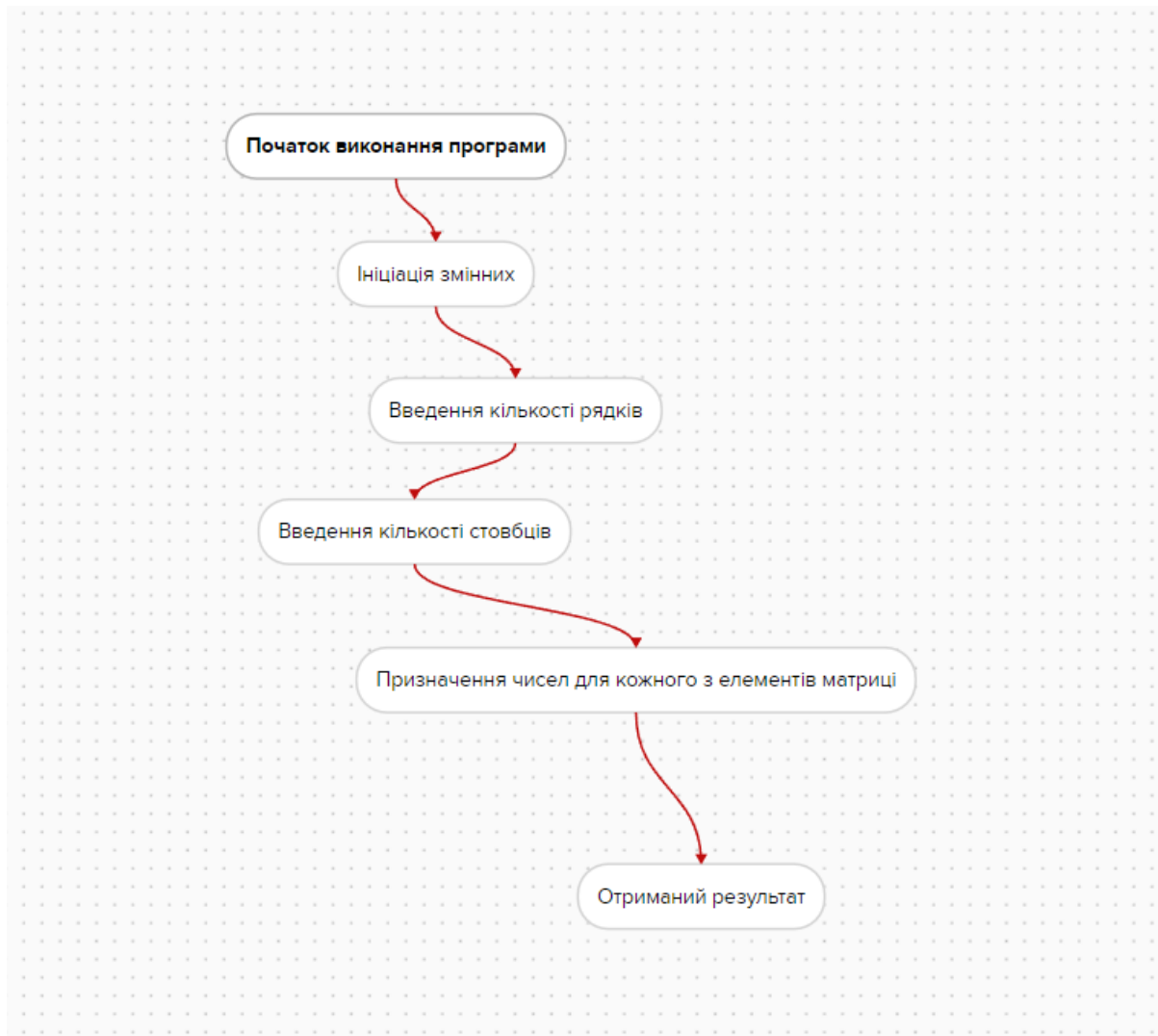
## ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі Matrix20

Вхідні дані: 1) максимальний розмір  $M = N = 20$ , ціле, константа; 2) кількість рядків  $row$ , ціле, 2..20 3) кількість стовпців  $col$ , ціле, 2..20 4) цілочисельний двовимірний масив  $matr1$ .

Вихідні дані: 1) елементи матриці; 2) сума елементів діагоналі  $sm$ , ціле.



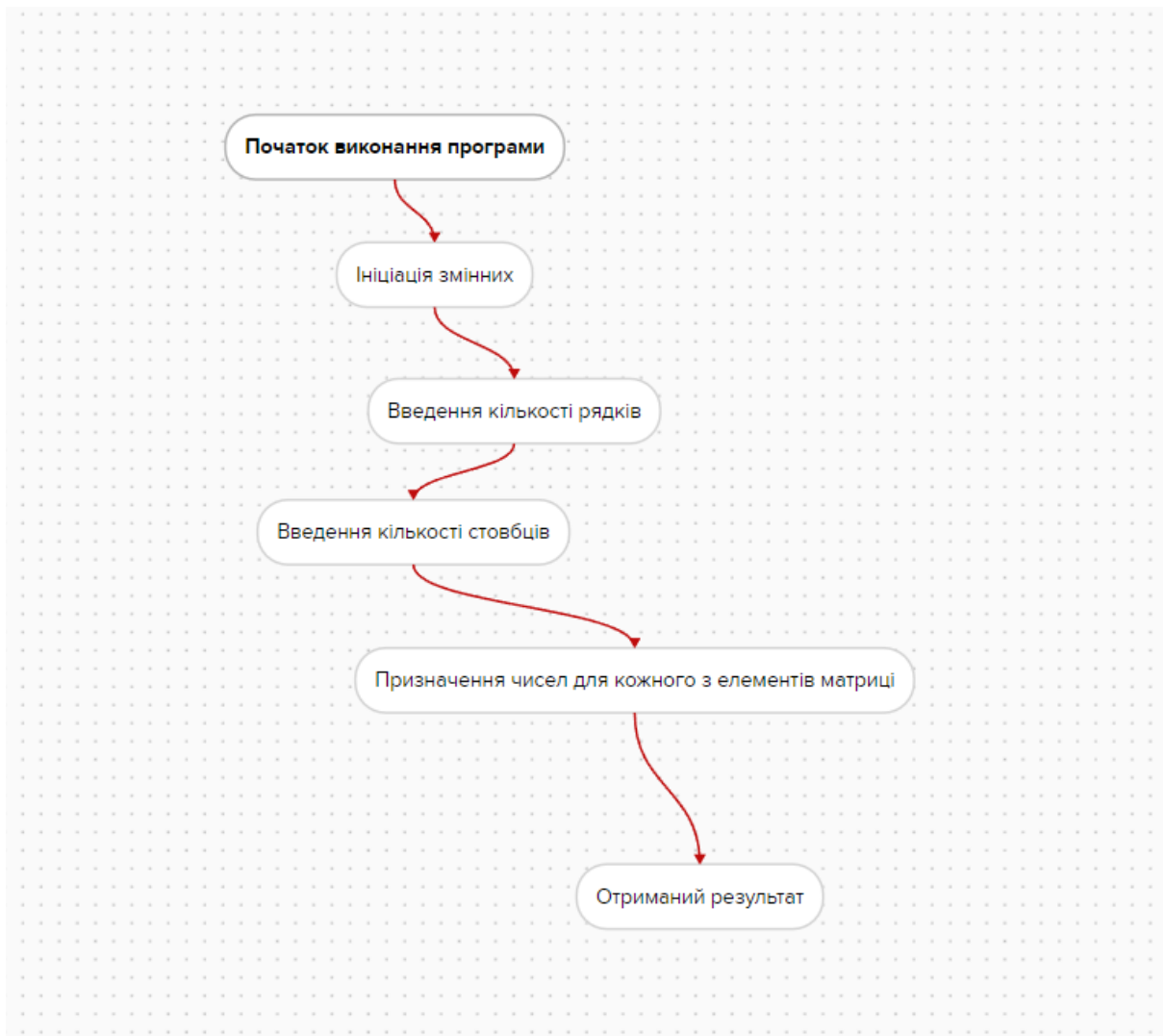
Малюнок 1

Завдання 2.

Вирішення задачі Matrix53

Вхідні дані: 1) максимальний розмір  $M = N = 20$ , ціле, константа; 2) кількість рядків  $row$ , ціле,  $2..20$  3) кількість стовпців  $col$ , ціле,  $2..20$  4) цілочисельний двовимірний масив  $matr1$ .

Вихідні дані: 1) елементи матриці; 2) сума елементів діагоналі  $sm$ , ціле.



## ВИСНОВКИ

Ціль цієї лабораторної роботи - показати можливість використання матриць в мові c++

## ДОДАТОК А

### Лістинг коду програми

```
/******
```

Online C++ Compiler.

Code, Compile, Run and Debug C++ program online.

Write your code in this editor and press "Run" button to compile and execute it.

```
*****/
```

Завдання 1

```
#include <iostream>
```

```
#include <vector>
```

```
using namespace std;
```

```
const int M = 20;
```

```
const int N = 20;
```

```
int main() {
```

```
    // Введення кількості рядків та стовпців
```

```
    int row, col;
```

```
    cout << "Введіть кількість рядків (від 2 до 20): ";
```

```
    cin >> row;
```

```
    cout << "Введіть кількість стовпців (від 2 до 20): ";
```

```
    cin >> col;
```

```
    // Перевірка на допустимість розмірів матриці
```

```
    if (row < 2 || row > M || col < 2 || col > N) {
```

```
        cout << "Неправильні розміри матриці!" << endl;
```

```
        return 1;
```

```
    }
```

```
    // Ініціалізація матриці та суми елементів діагоналі
```

```
    int matr1[M][N];
```

```
    int sm = 0;
```

```
    // Введення елементів матриці та обчислення діагональних елементів
```

```
    cout << "Введіть елементи матриці через пробіл:" << endl;
```

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

```
        cout << "Рядок " << i + 1 << ": ";
```

```
        for (int j = 0; j < col; ++j) {
```

```

        cin >> matr1[i][j];

        // Обчислення суми елементів діагоналі
        if (i == j)
            sm += matr1[i][j];
    }
}

// Виведення елементів матриці
cout << "Елементи матриці:" << endl;
for (int i = 0; i < row; ++i) {
    for (int j = 0; j < col; ++j) {
        cout << matr1[i][j] << " ";
    }
    cout << endl;
}

// Виведення суми елементів діагоналі
cout << "Сума елементів діагоналі: " << sm << endl;

return 0;
}

```

## Завдання 2

```

#include <iostream>
#include <vector>
using namespace std;

const int M = 20;
const int N = 20;

int main() {
    // Введення розмірів матриці

```

```

int row, col;

cout << "Введіть кількість рядків (від 2 до 20): ";

cin >> row;

cout << "Введіть кількість стовпців (від 2 до 20): ";

cin >> col;


// Перевірка на допустимість розмірів матриці
if (row < 2 || row > M || col < 2 || col > N) {

    cout << "Неправильні розміри матриці!" << endl;

    return 1;

}


// Ініціалізація матриці
vector<vector<int>> matr(row, vector<int>(col));


// Введення елементів матриці
cout << "Введіть елементи матриці:" << endl;

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

    cout << "Рядок " << i + 1 << ": ";

    for (int j = 0; j < col; ++j) {

        cin >> matr[i][j];

    }

}


// Пошук індексу стовпця з першим додатнім елементом
int first_positive_col = -1;

for (int j = 0; j < col; ++j) {

    bool has_positive = false;

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

        if (matr[i][j] > 0) {

            has_positive = true;

```

```

        break;
    }
}
if (has_positive) {
    first_positive_col = j;
    break;
}
}

```

// Пошук останнього стовпця з додатніми елементами

```

int last_positive_col = -1;
for (int j = col - 1; j >= 0; --j) {
    bool has_positive = false;
    for (int i = 0; i < row; ++i) {
        if (matr[i][j] > 0) {
            has_positive = true;
            break;
        }
    }
    if (has_positive) {
        last_positive_col = j;
        break;
    }
}

```

// Обмін стовпців, якщо знайдено відповідні стовпці

```

if (first_positive_col != -1 && last_positive_col != -1 && first_positive_col != last_positive_col) {
    for (int i = 0; i < row; ++i) {
        swap(matr[i][first_positive_col], matr[i][last_positive_col]);
    }
}

```



```
// Виведення матриці
cout << "Матриця після обміну стовпців:" << endl;
for (int i = 0; i < row; ++i) {
    for (int j = 0; j < col; ++j) {
        cout << matr[i][j] << " ";
    }
    cout << endl;
}

return 0;
}
```