

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 мовою C ++»

XAI.301.174. 319.23ЛР

Виконав студент гр. 319

Шаньгин А.А  
(підпис, дата) (П.І.Б.)

Перевірів

                     к.т.н., доц. Олена ГАВРИЛЕНКО  
(підпис, дата) (П.І.Б.)

## МЕТА РОБОТИ

Вивчити теоретичний матеріал з основ представлення двовимірних масивів (матриць) у мові C ++ і реалізувати декларацію, введення з консолі, обробку і виведення в консоль матриць мовою 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.

<b>Matrix35</b>	Дана цілочисельна матриця розміру $M \times N$ . Знайти номер першого з її стовпців, що містять тільки непарні числа. Якщо таких стовпців немає, то вивести 0.
-----------------	--

Завдання 2. Перетворити матрицю відповідно до свого варіанту завдання, розмір матриці і його елементи ввести з консолі. Вивести результати у консоль.

<b>Matrix62</b>	Дана матриця розміру $M \times N$ і ціле число $K$ ( $1 \leq K \leq N$ ). Видалити стовпець матриці з номером $K$ .
-----------------	---

## ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі Matrix35.

Вхідні дані (ім'я, опис, тип, обмеження):

M - кількість рядків матриці, дійсний тип,  $>0$ ;

N - кількість стовпців матриці, дійсний тип,  $>0$ ;

matrix[i][j] – елемент матриці в рядку i, стовпці j, дійсний тип,  $0 \leq \text{matrix}[i][j] \leq 100$ .

Вихідні дані (ім'я, опис, тип):

count – номер стовпця з непарними числами, дійсний тип,  $0 \leq \text{count} \leq N$

Алгоритм вирішення зображений на рис.1

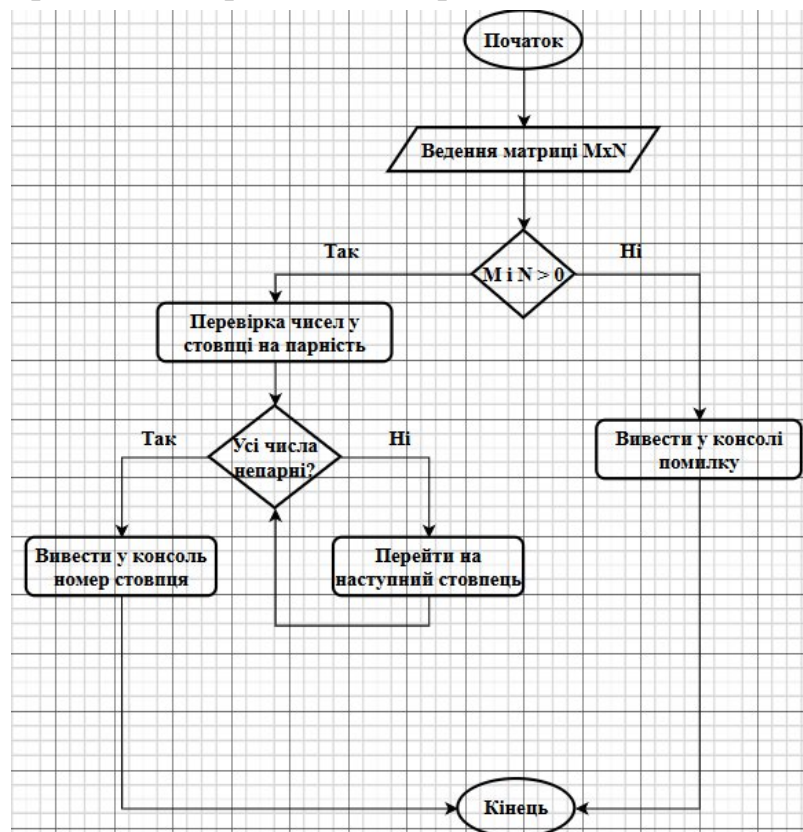


Рис.1 Алгоритм вирішення задачі Matrix35

Завдання 2.

Вирішення задачі Matrix62.

Вхідні дані (ім'я, опис, тип, обмеження):

M - кількість рядків матриці, дійсний тип,  $>0$ ;

N - кількість стовпців матриці, дійсний тип,  $>0$ ;

$matrix[i][j]$  – елемент матриці в рядку  $i$ , стовпці  $j$ , дійсний тип,  $0 \leq matrix[i][j] \leq 100$ .

Вихідні дані (ім'я, опис, тип):

count – нова матриця, дійсний тип.

Алгоритм вирішення на рис.2

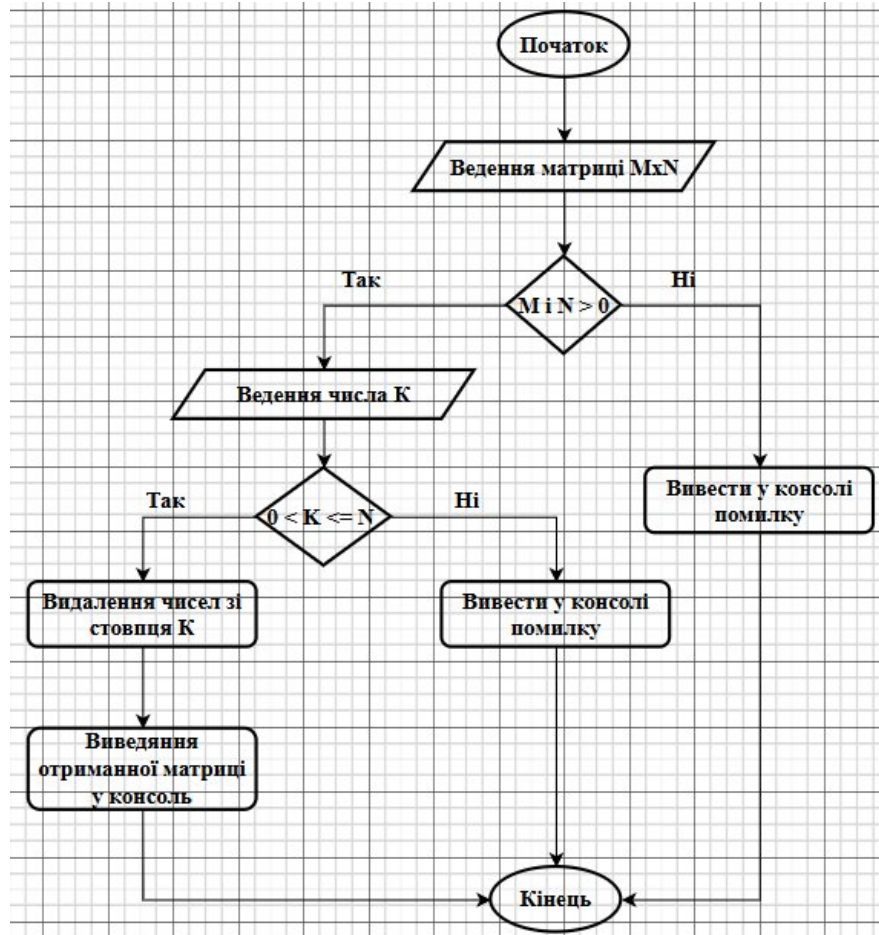


Рис.1 Алгоритм вирішення задачі Matrix62

Лістинг коду вирішення задач Matrix35 і Matrix62 наведено в дод. А (стор. 5). Екран роботи програми показаний на рис. Б.1.

## ВИСНОВКИ

У ході лабораторної роботи було реалізовано два алгоритми обробки матриць: підрахунок рядків з однаковими множинами елементів та обмін лівої і правої половин матриці. Отримані результати підтвердили ефективність використання множин для порівняння рядків та оптимальність in-place алгоритму для обміну половин. Робота дозволила закріпити навички роботи з двовимірними масивами та STL-контейнерами для вирішення практичних задач обробки даних.

## ДОДАТОК А

### Лістинг коду програми

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

// Функція для виводу матриці на екран
void printMatrix(const vector<vector<int>>& matrix) {
    for (const auto& row : matrix) {
        for (int element : row) {
            cout << element << " ";
        }
        cout << endl;
    }
}

// Завдання 1: Знайти номер першого стовпця, містить лише непарні числа
int findFirstOddColumn(const vector<vector<int>>& matrix) {
    int m = matrix.size(); // Кількість строк
    if (m == 0) return 0; // Перевірка на пусту матриці

    int n = matrix[0].size(); // Кількість стовпців

    // Перевірка кожного стовпця
    for (int j = 0; j < n; j++) {
        bool allOdd = true;

        // Перевірка, чи є всі елементи стовпця непарними
        for (int i = 0; i < m; i++) {
            if (matrix[i][j] % 2 == 0) { // Якщо знайдено парне число
                allOdd = false;
                break;
            }
        }

        // Якщо всі числа у стовпці непарні, повертаємо номер стовпця (с 1)
        if (allOdd) {
            return j + 1; // Повертаємо номер стовпця (нумерація с 1)
        }
    }

    // Якщо не знайдено стовпця з тільки непарними числами
    return 0;
}
```

```

// Завдання 2: Видалити стовпець K із матриці
vector<vector<int>> deleteColumn(const vector<vector<int>>& matrix, int k) {
    int m = matrix.size();    // Кількість строк
    if (m == 0) return {};    // Перевірка на пусту матрицю

    int n = matrix[0].size(); // Кількість стовпців

    // Перевірка, чи знаходиться K у допустимих межах
    if (k < 1 || k > n) {
        cout << "Error: K must be between 1 and " << n << endl;
        return matrix;
    }

    // Створюємо нову матрицю з кількістю стовпців на 1 менше
    vector<vector<int>> result(m, vector<int>(n - 1));

    // Копируємо елементи із ісходної матриці, пропускаючи стовпець K
    for (int i = 0; i < m; i++) {
        for (int j = 0, newCol = 0; j < n; j++) {
            if (j != k - 1) { // k-1, так як нумерація стовбців у задачі з 1, а
                // в C++ з 0
                result[i][newCol++] = matrix[i][j];
            }
        }
    }

    return result;
}

int main() {

    int choice;
    bool continueProgram = true;

    while (continueProgram) {
        cout << "\n=== PROGRAM MENU ===" << endl;
        cout << "Select a task:\n";
        cout << "1 - Find the number of the first column with odd numbers\n";
        cout << "2 - Remove column K from the matrix\n";
        cout << "0 - Exit the program\n";
        cout << "Your choice: ";
        cin >> choice;

        if (choice == 0) {
            continueProgram = false;
        }
    }
}

```

```

        cout << "The program is complete." << endl;
        continue;
    }

    if (choice != 1 && choice != 2) {
        cout << "Invalid task selection! Please select 1, 2 or 0 to exit.\n"
<< endl;
        continue;
    }

    int m, n;
    cout << "Enter the matrix dimensions (M and N): ";
    cin >> m >> n;

    // Перевірка коректності введених розмірів
    if (m <= 0 || n <= 0) {
        cout << "Error: matrix dimensions must be positive numbers" << endl;
        continue;
    }

    // Створення матриці
    vector<vector<int>> matrix(m, vector<int>(n));

    cout << "Enter the matrix elements:" << endl;
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrix[i][j];
        }
    }

    cout << "\nOriginal matrix:" << endl;
    printMatrix(matrix);

    if (choice == 1) {
        // Завдання 1: Пошук першого стовпця з непарними числами
        int result = findFirstOddColumn(matrix);
        cout << "\nThe number of the first column containing only odd
numbers: " << result;
        if (result == 0) {
            cout << " (there are no such columns)";
        }
        cout << endl;
    }
    else if (choice == 2) {
        // Завдання 2: Видалення стовпця K
        int k;

```



```

    cout << "\nEnter column number K to delete (1 <= K <= " << n << "):
";

    cin >> k;

    // Перевірка, чи знаходиться K у допустимих межах
    if (k < 1 || k > n) {
        cout << "Error: K must be between 1 and" << n << endl;
        continue;
    }

    vector<vector<int>> result = deleteColumn(matrix, k);

    cout << "\nMatrix after removing a column " << k << ":" << endl;
    printMatrix(result);
}

cout << "\nPress Enter to continue...";
cin.ignore(); // Очищення буфера
cin.get();    // Очікування натискання клавіш
}

return 0;
}

```

## ДОДАТОК Б

### Скріншот вікна виконання програми

```
=== PROGRAM MENU ===
Select a task:
1 - Find the number of the first column with odd numbers
2 - Remove column K from the matrix
0 - Exit the program
Your choice: 1
Enter the matrix dimensions (M and N): 5 4
Enter the matrix elements:
1 2 3 4
5 6 7 8
12 13 11 10
99 98 97 96
10 11 13 14

Original matrix:
1 2 3 4
5 6 7 8
12 13 11 10
99 98 97 96
10 11 13 14

The number of the first column containing only odd numbers: 3

Press Enter to continue...

=== PROGRAM MENU ===
Select a task:
1 - Find the number of the first column with odd numbers
2 - Remove column K from the matrix
0 - Exit the program
Your choice: 2
Enter the matrix dimensions (M and N): 5 4
Enter the matrix elements:
1 2 3 4
5 6 7 8
12 13 11 10
99 98 97 96
10 11 13 14

Original matrix:
1 2 3 4
5 6 7 8
12 13 11 10
99 98 97 96
10 11 13 14

Enter column number K to delete (1 <= K <= 4): 3

Matrix after removing a column 3:
1 2 4
5 6 8
12 13 10
99 98 96
10 11 14

Press Enter to continue...

=== PROGRAM MENU ===
Select a task:
1 - Find the number of the first column with odd numbers
2 - Remove column K from the matrix
0 - Exit the program
Your choice: 0
The program is complete.
```

Рисунок Б.1 – Екран виконання програми для вирішення завдання Matrix35 і Matrix62.