МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 7

з дисципліни «Алгоритмізація та програмування»

Тема: "Реалізація алгоритмів обробки двовимірних масивів на мові С ++"

ХАІ.301.174.319.7 ЛР

Виконав студент гр. 319

Шамарін Данило

(підпис, дата) (П.І.Б.)

Перевірив

к.т.н., доц. Олена ГАВРИЛЕНКО

(підпис, дата) (П.І.Б.)

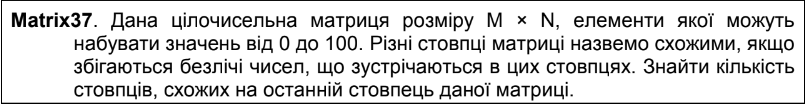
2025

МЕТА РОБОТИ

Вивчити теоретичний матеріал з основ представлення двовимірних масивів (матриць) на мові С ++ 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. Варіанти завдань представлено в табл.1.



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



ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі matrix 37. Дод А, Дод Б, A.1

Вхідні дані:

m: кількість рядків у матриці (ціле число)

n: кількість стовпців у матриці (ціле число)

matrix: двовимірний масив розміром m на n, що містить цілі числа

Обмеження:

1 <= m <= M (константа, визначена до початку програми)

1 <= n <= N (константа, визначена до початку програми)

Вихідні дані:

1. Ім'я: програма для підрахунку кількості стовпців матриці, в яких кожен елемент першого рядка збігається з елементом останнього рядка.
2. Опис: програма приймає від користувача кількість рядків та стовпців матриці, заповнює її елементами та рахує кількість стовпців, у яких кожен елемент першого рядка збігається з елементом останнього рядка.
3. Тип: консольний додаток.

Алгоритм виконання:

1. Оголошення змінних m та n.
2. Виведення повідомлень для проханням ввести кількість рядків та стовпців матриці.
3. Введення значень m та n.
4. Оголошення двомірного масиву matrix розміром m на n.
5. Заповнення матриці значеннями, введеними користувачем.

Завдання 2. Вирішення задачі matrix 64. A, Дод Б, A 2

Вхідні дані:

m: кількість рядків матриці, ціле число, обмеження: m <= M.

n: кількість стовпців матриці, ціле число, обмеження: n <= N.

Обмеження:

1. Для коректного введення матриці, кожен елемент матриці повинен бути цілим числом.

Вихідні дані:

1. min\_val: найменший елемент матриці, ціле число.
2. min\_col: номер стовпця, у якому знаходиться найменший елемент матриці, ціле число.

Алгоритм виконання:

1. Оголошення змінних m, n, min\_val, min\_col типу int.
2. Виведення запиту на введення кількості рядків та стовпців матриці з обмеженнями.
3. Зчитування кількості рядків та стовпців матриці з клавіатури.
4. Оголошення матриці розміром m на n типу int.
5. Виклик функції input\_matrix, яка заповнює матрицю значеннями, введеними користувачем.
6. Виклик функції print\_matrix, яка виводить матрицю на екран.
7. Виклик функції find\_min, яка знаходить найменший елемент матриці та його стовпець.

ВИСНОВОК

У процесі лабораторної роботи ми вивчили теоретичні матеріали граматики мови C++, вивчили теоретичний матеріал з основ представлення двовимірних масивів на мові С ++ і реалізувати оголошення, введення з консолі, обробку і виведення в консоль матриць на мові C ++ в середовищі Visual Studio

ДОДАТОК А

Лістинг коду програми до всіх задач

Лістинг коду

#include <iostream>

#include <cmath>

using namespace std;

const int M = 20, N = 20; // максимальне кількість рядків стовпців

void matrix37();

void matrix64();

int main()

{

int menu;

do

{

cout << "Task number (0 for exit):";

cin >> menu;

switch (menu)

{ // перемикання між завданнями

case 1:

matrix37();

break; // Завдання 1

case 2:

matrix64();

break; // Завдання 2

}

} while (menu != 0);

system("pause");

return 0;

}

void input\_matrix(int matrix[M][N], int rows, int cols)

{

cout << "Enter the size matrix " << rows << " x " << cols << ":" << endl;

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

{

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

{

cin >> matrix[i][j];

}

}

}

// функція виведення масиву

void print\_matrix(int matrix[M][N], int rows, int cols)

{

cout << "Size matrix " << rows << " x " << cols << ":" << endl;

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

{

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

{

cout << matrix[i][j] << " ";

}

cout << endl;

}

}

void find\_min(int matrix[M][N], int rows, int cols, int& min\_val, int& min\_col)

{

min\_val = matrix[0][0];

min\_col = 0;

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

{

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

{

if (matrix[i][j] < min\_val)

{

min\_val = matrix[i][j];

min\_col = j;

}

}

}

}

void find\_max(int matrix[M][N], int rows, int cols, int& max\_val, int& max\_col)

{

max\_val = matrix[0][0];

max\_col = 0;

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

{

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

{

if (matrix[i][j] > max\_val)

{

max\_val = matrix[i][j];

max\_col = j;

}

}

}

}

void matrix37()

{

int m, n;

cout << "Enter the number of lines (max " << M << "): ";

cin >> m;

cout << "Enter the number of columns (max " << N << "): ";

cin >> n;

int matrix[M][N];

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

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

cin >> matrix[i][j];

}

}

int count = 0;

for (int j = 0; j < n - 1; j++) {

bool similar = true;

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

bool found = false;

for (int k = 0; k < m; k++) {

if (matrix[i][j] == matrix[k][n - 1]) {

found = true;

break;

}

}

if (!found) {

similar = false;

break;

}

}

if (similar) {

count++;

}

}

cout << "Number of columns: " << count << endl;

}

void matrix64()

{

int m, n;

cout << "Enter the number of lines (max " << M << "): ";

cin >> m;

cout << "Enter the number of columns (max " << N << "): ";

cin >> n;

int matrix[M][N];

input\_matrix(matrix, m, n);

print\_matrix(matrix, m, n);

int min\_val, min\_col;

find\_min(matrix, m, n, min\_val, min\_col);

cout << "Minimum element: " << min\_val << ", located in the column " << min\_col << endl;

}

ДОДАТОК Б

Скріншоти вікна виконання програми

Рисунок Б.1 – Екран виконання програми для вирішення завдання 1 matrix 34

(A 1)

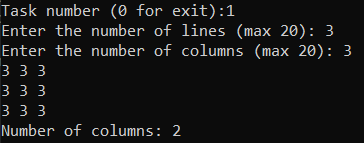


Рисунок Б.2 – Екран виконання програми для вирішення завдання 2 matrix 64 (A 2)

