**Лабораторна робота №4**

З дисципліни «Структури даних та алгоритми»

Алгоритми лінійного пошуку

Виконав: Землянський Едуард

Група: КВ-22

**Мета лабораторної роботи**

Метою лабораторної роботи №4. «Алгоритми лінійного пошуку» є засвоєння теоретичного та набуття практичних навичок рішення задачі пошуку заданої категорії елементів заидопомогою різних алгоритмів методу лінійного пошуку у двовимірних масивах.

**Постановка задачі**

1. Написати програму розв’язання задачі пошуку (за варіантом)у двовимірному масиві (матриці) одним з алгоритмів методу лі-нійного пошуку.
2. Розміри матриці m та n взяти самостійно у межах від 7 до 10.
3. Виконати тестування та налагодження програми накомп’ютері. При тестуванні програми необхідно підбирати такі вхі-дні набори початкових значеннь матриці, щоб можна було легковідстежити коректність виконання пошуку і ця коректність була бпротестована для всіх можливих випадків. З метою тестування до-зволяється використовувати матриці меншого розміру.

**Текст програми**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <time.h>

void fill\_matrix(int rows, int columns, int matrix[rows][columns]){

    for (int i = 0; i <  rows; i++){

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

            matrix[i][j] = rand()%10;

        }

    }

}

void display\_matrix(int rows, int columns, int matrix[rows][columns]){

    for (int i = 0; i < rows; i++){

        printf("\n");

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

            printf("%d ", matrix[i][j]);

        }

    }

}

void find\_min(int rows, int columns, int matrix[rows][columns]){

    int min = matrix[0][0];

    int coord1, coord2;

    for (int r = 0; r <  rows; r++){

        for (int c = 0; c <  columns; c++){

            if (min >= matrix[c][r]){

                min = matrix[c][r];

                coord1 = c;

                coord2 = r;

            }

        }

    }

    printf("\nMinimum element: %d, coords: (%d, %d)", min, coord1+1, coord2+1);

}

int main(){

    srand(time(0));

    int matrix[7][7];

    fill\_matrix(7,7, matrix);

    printf("\nMatrix:");

    display\_matrix(7, 7, matrix);

    find\_min(7, 7, matrix);

    return 0;

}

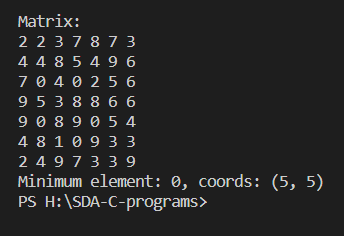
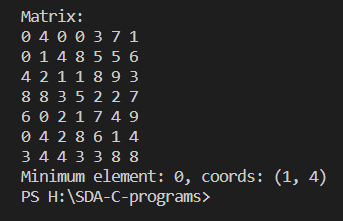
// Варіант 12

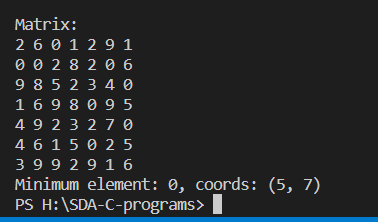
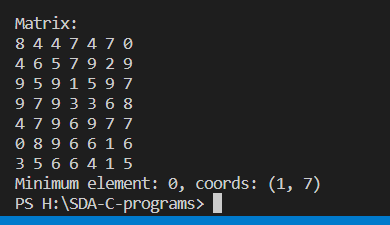
// Задано матрицю цілих чисел A[m,n]. При обході матриці по

// стовпчиках знайти в ній останній мінімальний елемент і його

// місцезнаходження (координати).

**Тестування, результати для різних вхідних даних:**

**** ****

**** ****