

**Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря
Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки**

Лабораторна робота №1.5
з дисципліни
«Алгоритми і структури даних»

Виконала:

Студентка групи ІМ-41
Куц Анна Василівна номер
у списку групи: 11

Перевірила:

Молчанова А. А.

Київ 2024

Постановка задачі

1. Написати програму розв'язання задачі пошуку (за варіантом) у двовимірному масиві (матриці) одним з алгоритмів методу лінійного пошуку.
2. Розміри матриці *m* та *n* взяти самостійно у межах від 7 до 10.
3. Виконати тестування та налагодження програми на комп'ютері. При тестуванні програми необхідно підбирати такі вхідні набори початкових значень матриці, щоб можна було легко відстежити коректність виконання пошуку і ця коректність була б протестована для всіх можливих випадків. З метою тестування дозволяється використовувати матриці меншого розміру.

Завдання

Варіант 11

Задано матрицю дійсних чисел $A[n,n]$. У побочній діагоналі матриці знайти перший максимальний і останній мінімальний елементи, а також поміняти їх місцями.

Текст програми

```
#include <stdio.h>

int main()
{
    int n;

    printf("Enter n of rows and columns: \n");
    scanf("%d", &n);

    int matrix[n][n];

    if(n<1) {
        printf("Incorrect value\n");
        return 1;
    }
```

```

printf("Enter elements of matrix: \n");
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < n; j++) {
scanf("%d", &matrix[i][j]);
    }
}
printf("Your matrix: \n");
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
printf("%d ", matrix[i][j]);
        }
printf("\n");
    }

    int max = matrix[n-1][0];
int min = matrix[n-1][0];
int max_index = n-1;
int min_index = n-1;
for (int i = 0; i < n; i++) {
    int diag_element = matrix[n-i-1][i];
    if (diag_element > max)
    {
        max = diag_element;
        max_index = n-i-1;
    }
    if (diag_element <= min)
    {
        min = diag_element;
        min_index = n-i-1;
    }
}

```

```

    }
}

printf("Your max element:\n%d\n", max);
printf("Your min element:\n%d\n", min);

int temp = matrix[max_index][n - max_index - 1];
matrix[max_index][n - max_index - 1] = matrix[min_index][n
- min_index - 1];
matrix[min_index][n - min_index - 1] = temp;

printf("Changed matrix:\n");
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < n; j++)
    {
        printf("%d ", matrix[i][j]);
    }
    printf("\n");
}
return 0;
}

```

Тестування програми

```
"/Users/annkuts/University/АСД/Лабораторні/lab 1.5/lab 1.5"
```

```
Enter n of rows and columns:
```

```
3
```

```
Enter elements of matrix:
```

```
0 0 8
```

```
0 8 1
```

```
1 0 0
```

```
Your matrix:
```

```
0 0 8
```

```
0 8 1
```

```
1 0 0
```

```
Your max element:
```

```
8
```

```
Your min element:
```

```
1
```

```
Changed matrix:
```

```
0 0 8
```

```
0 1 1
```

```
8 0 0
```

```
Process finished with exit code 0
```

"/Users/annkuts/University/АСД/Лабораторні/lab 1.5/lab 1.5"

Enter n of rows and columns:

4

Enter elements of matrix:

-9 0 0 9
0 -9 9 0
0 -9 0 0
9 0 0 -9

Your matrix:

-9 0 0 9
0 -9 9 0
0 -9 0 0
9 0 0 -9

Your max element:

9

Your min element:

-9

Changed matrix:

-9 0 0 9
0 -9 9 0
0 9 0 0
-9 0 0 -9

Process finished with exit code 0

"/Users/annkuts/University/АСД/Лабораторні/lab 1.5/lab 1.5"

Enter n of rows and columns:

5

Enter elements of matrix:

```
9 0 0 0 -9
0 9 0 -9 0
0 0 9 0 0
0 -9 0 9 0
-9 0 0 0 9
```

Your matrix:

```
9 0 0 0 -9
0 9 0 -9 0
0 0 9 0 0
0 -9 0 9 0
-9 0 0 0 9
```

Your max element:

9

Your min element:

-9

Changed matrix:

```
9 0 0 0 9
0 9 0 -9 0
0 0 -9 0 0
0 -9 0 9 0
-9 0 0 0 9
```

Process finished with exit code 0

"/Users/annkuts/University/АСД/Лабораторні/lab 1.5/lab 1.5"

Enter n of rows and columns:

6

Enter elements of matrix:

9 0 0 0 0 -9
0 9 0 0 -9 0
0 0 9 -9 0 0
0 0 -9 9 0 0
0 -9 0 0 9 0
-9 0 0 0 0 9

Your matrix: |

9 0 0 0 0 -9
0 9 0 0 -9 0
0 0 9 -9 0 0
0 0 -9 9 0 0
0 -9 0 0 9 0
-9 0 0 0 0 9

Your max element:

-9

Your min element:

-9

Changed matrix:

9 0 0 0 0 -9
0 9 0 0 -9 0
0 0 9 -9 0 0
0 0 -9 9 0 0
0 -9 0 0 9 0
-9 0 0 0 0 9

Process finished with exit code 0

"/Users/annkuts/University/АСД/Лабораторні/lab 1.5/lab 1.5"

Enter n of rows and columns:

7

Enter elements of matrix:

```
9 0 0 0 0 0 -9
0 9 0 0 0 -9 0
0 0 9 0 -9 0 0
0 0 0 9 0 0 0
0 0 -9 0 9 0 0
0 -9 0 0 0 9 0
-9 0 0 0 0 0 9
```

Your matrix:

```
9 0 0 0 0 0 -9
0 9 0 0 0 -9 0
0 0 9 0 -9 0 0
0 0 0 9 0 0 0
0 0 -9 0 9 0 0
0 -9 0 0 0 9 0
-9 0 0 0 0 0 9
```

Your max element:

9

Your min element:

-9

Changed matrix:

```
9 0 0 0 0 0 9
0 9 0 0 0 -9 0
0 0 9 0 -9 0 0
0 0 0 -9 0 0 0
0 0 -9 0 9 0 0
0 -9 0 0 0 9 0
-9 0 0 0 0 0 9
```

Process finished with exit code 0

```
"/Users/annkuts/University/АСД/Лабораторні/lab 1.5/lab 1.5"
```

```
Enter n of rows and columns:
```

```
4
```

```
Enter elements of matrix:
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

```
Your matrix:
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

```
Your max element:
```

```
0
```

```
Your min element:
```

```
0
```

```
Changed matrix:
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

```
Process finished with exit code 0
```

Висновок

Завдяки виконанню лабораторної роботи №1.5 я засвоїла теоретичний матеріал та набула практичних навичок рішень задач пошуку заданої категорії елементів за допомогою різних алгоритмів методу лінійного пошуку у двовимірних масивах. Було створено програму для знаходження максимального і мінімального елементу в побічній діагоналі матриці та зміни їх місцями. Складність алгоритму є $O(n^2)$, оскільки при створенні програми я використовувала вкладені цикли. Окрім цього, за допомогою матеріалу, викладеного на лекціях, я зуміла застосувати метод лінійного пошуку, який знаходить відповідне значення max та min у моїй програмі.