

**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 №3
з дисципліни
«Алгоритми і структури даних»

Виконав:

студент групи ІМ-44
Мундурс Нікіта Юрійович
номер у списку групи: 16

Перевірила:

Молчанова А. А.

Київ 2024

Завдання

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.

Варіант 16:

Задано матрицю дійсних чисел $A[m,n]$. У кожному стовпчику матриці знайти перший від'ємний елемент і його місцезнаходження (координати).

Текст програми

```
#include <stdio.h>
```

```
#include <stdbool.h>
```

```
#define M 10
```

```
#define N 8
```

```
void printMatrix(const double matrix[][N], int rows, int cols) {
```

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

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

```
            printf("%.2lf\t", matrix[i][j]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
}
```

```
int main(const int argc, const char* argv[]) {
```

```
    const double matrix[M][N] = {
```

```
        { -8.37, 3.79, 4.93, 3.31, 0.04, -8.31, 5.83, 1.01 },
```

```
        { 7.33, 5.35, -6.19, 0.54, 2.31, -4.05, 2.64, 2.54 },
```

```
        { -7.47, -1.39, -2.57, 2.88, 8.03, 6.88, 0.61, 3.02 },
```

```
        { -8.85, -9.88, 7.63, -1.50, 2.70, 2.80, -9.33, 5.97 },
```

```
        { 2.09, -4.18, 4.82, -5.82, 2.57, -6.83, 5.09, 0.21 },
```

```
        { 6.14, 9.96, 7.74, 1.23, -7.13, 5.35, -1.25, 7.62 },
```

```
        { -7.53, 7.17, -1.72, 2.13, -1.62, 4.55, -2.47, 0.99 },
```

```
        { -4.55, 6.89, -3.56, 2.56, 7.62, 5.56, 0.19, 1.01 },
```

```
        { 7.33, 5.35, -6.19, 0.54, 2.31, -4.05, 2.64, 2.59 },
```

```
        { 2.09, -4.18, 4.82, -5.82, 2.57, -6.83, 5.09, -0.21 },
```

```
    };
```

```
printMatrix(matrix, M, N);
```

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

```
bool finded = false;
```

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

```
const double item = matrix[j][i];
```

```
if (item >= 0) continue;
```

```
finded = true;
```

```
printf("Item: %.2lf, position = (%d, %d)\n", item, j, i);
```

```
break;
```

```
}
```

```
if (!finded) printf("There is not negative values in row\n");
```

```
}
```

```
return 0;
```

```
}
```

Тестування програми

```
PS D:\programming\code\lab3> gcc -o main.exe main.c
PS D:\programming\code\lab3> .\main.exe
-8.37  3.79  4.93  3.31  0.04  -8.31  5.83  1.01
7.33   5.35  -6.19  0.54  2.31  -4.05  2.64  2.54
-7.47  -1.39  -2.57  2.88  8.03  6.88  0.61  3.02
-8.85  -9.88  7.63  -1.50  2.70  2.80  -9.33  5.97
2.09   -4.18  4.82  -5.82  2.57  -6.83  5.09  0.21
6.14   9.96  7.74  1.23  -7.13  5.35  -1.25  7.62
-7.53  7.17  -1.72  2.13  -1.62  4.55  -2.47  0.99
-4.55  6.89  -3.56  2.56  7.62  5.56  0.19  1.01
7.33   5.35  -6.19  0.54  2.31  -4.05  2.64  2.59
2.09   -4.18  4.82  -5.82  2.57  -6.83  5.09  -0.21
Item: -8.37, position = (0, 0)
Item: -1.39, position = (2, 1)
Item: -6.19, position = (1, 2)
Item: -1.50, position = (3, 3)
Item: -7.13, position = (5, 4)
Item: -8.31, position = (0, 5)
Item: -9.33, position = (3, 6)
Item: -0.21, position = (9, 7)
```

```
PS D:\programming\code\lab3> gcc -o main.exe main.c
PS D:\programming\code\lab3> .\main.exe
-8.37  3.79  4.93  3.31  0.04  -8.31  5.83  1.01
7.33   5.35  -6.19  0.54  2.31  -4.05  2.64  2.54
-7.47  -1.39  -2.57  2.88  8.03  6.88  0.61  3.02
-8.85  -9.88  7.63  1.50  2.70  2.80  -9.33  5.97
2.09   -4.18  4.82  5.82  2.57  -6.83  5.09  0.21
6.14   9.96  7.74  1.23  -7.13  5.35  -1.25  7.62
-7.53  7.17  -1.72  2.13  -1.62  4.55  -2.47  0.99
-4.55  6.89  -3.56  2.56  7.62  5.56  0.19  1.01
7.33   5.35  -6.19  0.54  2.31  -4.05  2.64  2.59
2.09   -4.18  4.82  5.82  2.57  -6.83  5.09  0.21
Item: -8.37, position = (0, 0)
Item: -1.39, position = (2, 1)
Item: -6.19, position = (1, 2)
There is not negative values in row
Item: -7.13, position = (5, 4)
Item: -8.31, position = (0, 5)
Item: -9.33, position = (3, 6)
There is not negative values in row
```

```
PS D:\programming\code\lab3> gcc -o main.exe main.c
PS D:\programming\code\lab3> .\main.exe
8.37    3.79    4.93    3.31    0.04    8.31    5.83    1.01
7.33    5.35    6.19    0.54    2.31    4.05    2.64    2.54
7.47    1.39    2.57    2.88    8.03    6.88    0.61    3.02
8.85    9.88    7.63    1.50    2.70    2.80    9.33    5.97
2.09    4.18    4.82    5.82    2.57    6.83    5.09    0.21
6.14    9.96    7.74    1.23    7.13    5.35    1.25    7.62
7.53    7.17    1.72    2.13    1.62    4.55    2.47    0.99
4.55    6.89    3.56    2.56    7.62    5.56    0.19    1.01
7.33    5.35    6.19    0.54    2.31    4.05    2.64    2.59
2.09    4.18    4.82    5.82    2.57    6.83    5.09    0.21
There is not negative values in row
There is not negative values in row
There is not negative values in row
There is not negative values in row
There is not negative values in row
There is not negative values in row
There is not negative values in row
There is not negative values in row
```

Висновки

Протягом виконання лабораторної роботи я засвоїв теоретичний матеріал та набув практичні навички рішення задачі пошуку різних категорій елементів за допомогою алгоритмів методу лінійного пошуку у двумірному масиві.

Також я покращив навички алгоритмізації та програмування мовою C.