

**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 №5
з дисципліни
«Алгоритми і структури даних»

Виконав:

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

Перевірила:

Молчанова А. А.

Завдання:

1. Написати програму розв'язання задачі пошуку (за варіантом) у двовимірному масиві (матриці) методом двійкового пошуку. Алгоритм двійкового пошуку задається варіантом завдання.

2. Розміри матриці m та n взяти самостійно у межах від 7 до 10.

3. 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 $A[m,n]$. Окремо у кожному рядку матриці визначити присутність будь-якого з чисел діапазону $[0,5]$ і його місцезнаходження (координати) методом двійкового пошуку (Алгоритм №2), якщо елементи кожного рядка окремо впорядковані за незбільшенням.

Текст програми:

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

```
#define ROWS 7
```

```
#define COLS 10
```

```
int binarySearch(double row[], int cols, double min, double max, double*  
value) {
```

```
    int left = 0, right = cols - 1;
```

```
    int index = -1;
```

```
    while (left <= right) {
```

```
        int mid = left + (right - left) / 2;
```

```
        if (row[mid] >= min && row[mid] <= max) {
```

```
            index = mid;
```

```
            *value = row[mid];
```

```
            right = mid - 1;
```

```
        }
```

```
        else if (row[mid] < min) {
```

```
            right = mid - 1;
```

```
        }
```

```
        else {
```

```
            left = mid + 1;
```

```
        }
```

```
    }
```

```

    return index;
}

int main() {
    double matrix[ROWS][COLS] = {
        {10.0, 8.5, 5.0, 5.0, 4.0, 3.5, 3.1, 2.2, 1.5, 0.9},
        {15.0, 14.0, 12.0, 10.0, 5.0, 4.9, 4.5, 3.0, 2.0, 0.1},
        {50.0, 40.0, 30.0, 20.0, 20.0, 15.0, 13.3, 8.2, 8.1, 7.5},
        {4.9, 4.8, 4.7, 4.6, 4.5, 4.4, 4.3, 4.2, 4.1, 4.0},
        {8.5, 8.4, 7.5, 7.0, 6.5, 6.0, 5.9, 5.8, 5.6, 0.0},
        {6.0, 5.0, 4.8, 4.6, 4.4, 3.9, 3.5, 3.2, 2.8, 2.5},
        {9.9, 9.5, 8.5, 6.5, 5.6, 5.5, 3.5, 2.5, 1.5, 0.5},
    };

    double min = 0.0, max = 5.0;

    printf("Matrix:\n");
    for (int i = 0; i < ROWS; i++) {
        for (int j = 0; j < COLS; j++) {
            printf("%6.1f ", matrix[i][j]);
        }
        printf("\n");
    }
    printf("\n");
}

```

```
printf("Search results:\n");  
for (int i = 0; i < ROWS; i++) {  
    double value = 0.0;  
    int col = binarySearch(matrix[i], COLS, min, max, &value);  
    if (col != -1) {  
        printf("Row %d, Column %d, Value: %.1f\n", i, col, value);  
    } else {  
        printf("No element in range [%.1f, %.1f] found in row %d\n", min,  
max, i);  
    }  
}  
  
return 0;  
}
```

Тестування програми:

```
Matrix:
10.0  8.5  5.0  5.0  4.0  3.5  3.1  2.2  1.5  0.9
15.0  14.0 12.0 10.0  5.0  4.9  4.5  3.0  2.0  0.1
50.0  40.0 30.0 20.0 20.0 15.0 13.3  8.2  8.1  7.5
4.9   4.8  4.7  4.6  4.5  4.4  4.3  4.2  4.1  4.0
8.5   8.4  7.5  7.0  6.5  6.0  5.9  5.8  5.6  0.0
6.0   5.0  4.8  4.6  4.4  3.9  3.5  3.2  2.8  2.5
9.9   9.5  8.5  6.5  5.6  5.5  3.5  2.5  1.5  0.5
```

```
Search results:
Row 0, Column 2, Value: 5.0
Row 1, Column 4, Value: 5.0
No element in range [0.0, 5.0] found in row 2
Row 3, Column 0, Value: 4.9
Row 4, Column 9, Value: 0.0
Row 5, Column 1, Value: 5.0
Row 6, Column 6, Value: 3.5
```

```
Matrix:
110.0 98.5 95.1 85.6 76.0 73.5 53.1 42.2 41.5 10.9
15.0  14.0 12.0 9.0  8.0  7.9  6.2  6.2  6.0  5.1
50.0  40.0 30.0 20.0 20.0 15.0 13.3  8.2  8.1  7.5
55.9  54.8 44.7 44.6 34.5 34.4 24.3 24.2 14.1 14.0
8.5   8.4  7.5  7.0  6.5  6.0  5.9  5.8  5.6  5.1
67.0  13.0 12.8 11.6 10.4 9.9  8.5  7.2  6.8  5.5
9.9   9.5  8.5  6.5  5.6  5.5  5.5  5.5  5.3  5.1
```

```
Search results:
No element in range [0.0, 5.0] found in row 0
No element in range [0.0, 5.0] found in row 1
No element in range [0.0, 5.0] found in row 2
No element in range [0.0, 5.0] found in row 3
No element in range [0.0, 5.0] found in row 4
No element in range [0.0, 5.0] found in row 5
No element in range [0.0, 5.0] found in row 6
```

```
Matrix:
10.0  8.5  5.0  5.0  4.0  3.5  3.1  2.2  1.5  0.9
15.0  14.0 12.0 10.0  5.0  4.9  4.5  3.0  2.0  0.1
50.0  40.0 30.0 20.0 20.0  5.0  3.3  3.2  1.1  0.5
4.9   4.8  4.7  4.6  4.5  4.4  4.3  4.2  4.1  4.0
8.5   8.4  7.5  7.0  6.5  6.0  5.9  5.8  5.6  0.0
6.0   5.0  4.8  4.6  4.4  3.9  3.5  3.2  2.8  2.5
9.9   9.5  8.5  6.5  5.6  5.5  3.5  2.5  1.5  0.5
```

```
Search results:
Row 0, Column 2, Value: 5.0
Row 1, Column 4, Value: 5.0
Row 2, Column 5, Value: 5.0
Row 3, Column 0, Value: 4.9
Row 4, Column 9, Value: 0.0
Row 5, Column 1, Value: 5.0
Row 6, Column 6, Value: 3.5
```

Висновки. Протягом виконання цієї лабораторної роботи я засвоїв теоретичний матеріал та набув практичних навичок в роботі з алгоритмами двійкового пошуку у двовимірних масивах. Також я покращив свої навички алгоритмізації та вдосконалив програмування мовою С.