

**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 №2.1

з дисципліни
«Алгоритми і структури даних»

Виконав:

студент групи ІМ-32
Король Олександр Володимирович
номер у списку групи: 15

Перевірила:

Молчанова А. А.

Київ 2023

Постановка задачі

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 № 15

Задано матрицю дійсних чисел $A[m,n]$. Окремо у останньому рядку і першому стовпчику визначити присутність заданого дійсного числа X і його місцезнаходження (координати) методом двійкового пошуку (Алгоритм №1), якщо елементи цього рядка і стовпчика впорядковані за неменшенням.

Текст програми

```
#include <stdio.h>

int main() {

    int m = 8, n = 7, column = -1, row = -1;

    double x;
```

```

int low; // the start (first index) (~left)

int high; // the end (last index) (~right)

double Array[8][7] = {

    {1.1 ,1.2 ,1.3, 2.4, 2.5, 2.7, 3.7},

    {1.1, 3.2, 4.3, 4.4, 4.5, 5.1, 3.7},

    {1.1, 5.2, 5.3, 6.4, 6.4, 6.7, 6.7},

    {1.1, 7.2, 7.3, 7.4, 8.1, 8.1, 8.3},

    {1.1, 8.1, 9.2, 9.3, 9.4, 9.5, 10.4},

    {1.1, 10.2, 10.2, 10.2, 10.3, 11.1, 11.5},

    {1.1, 11.2, 12.3, 12.4, 12.4, 12.5, 13.6},

    {1.1, 2.1, 2.1, 2.1, 2.2, 2.2, 2.3},

};

for (int i = 0; i < 8; i++) {

    for (int j = 0; j < 7; j++) {

        printf("%2.11f ", Array[i][j]);

    }

    printf("\n");

}

printf("Enter X: ");

scanf("%lf", &x);

if (Array[m-1][0] == x) {

    // check element in the corner to see whether it is a searched
element or not

    column = 0;

    row = m-1;

} else {

    // if not, check whether it is lower than element in the corner or
not

```

```

        if (x < Array[m-1][0]) {

            // if so, then we should apply binary search only to first
column
            low = 0;

            high = m-2; // because we have already checked element in the
corner

            while (low <= high) {

                int midIndex = (high + low) / 2;

                double midValue = Array[midIndex][0];

                if (midValue == x) {

                    column = 0;

                    row = midIndex;

                    break;

                } else if (midValue < x) {

                    low = midIndex + 1;

                } else {

                    high = midIndex - 1;

                }

            }

        } else {

            // if not, then we should apply binary search only to last row

            low = 1; // because we have already checked element in the
corner

            high = n-1;

            while (low <= high) {

                int midIndex = (high + low) / 2;

                double midValue = Array[m-1][midIndex];

                if (midValue == x) {

                    column = midIndex;

                    row = m-1;

                    break;

```

```

        } else if (midValue < x) {

            low = midIndex + 1;

        } else {

            high = midIndex - 1;

        }

    }

}

}

if (row != -1) {

    printf("The pos of X = [%d][%d]", row , column);

} else {

    printf("Not founded in the first column and last row.");

}

return 0;

}

```

Результати тестування програми

Елемент один, і знаходиться посередині (в куті)

```

1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.3 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.5 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.7 11.2 12.3 12.4 12.4 12.5 13.6
1.8 2.1 2.1 2.1 2.2 2.2 2.3
Enter X:1.8
The pos of X = [7][0]
Process finished with exit code 0

```

3 елемента підряд (шуканий посередині, тобто в куті)

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.3 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.5 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.8 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:1.8
The pos of X = [7][0]
Process finished with exit code 0
```

Шуканий елемент має дубль зліва від нього

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.4 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.5 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.8 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:1.4
The pos of X = [3][0]
Process finished with exit code 0
```

Шуканий елемент має дубль справа від нього

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.3 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.4 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.8 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:1.4
The pos of X = [3][0]
Process finished with exit code 0
```

Шуканий елемент має дубль і зліва, і справа від нього

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.4 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.4 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.8 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:1.4
The pos of X = [3][0]
Process finished with exit code 0
```

Шуканого елемента немає в масиві

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.4 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.4 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.8 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:666
Not founded in the first column and last row.
Process finished with exit code 0
```

Шуканого елемента немає в першому стовпчику і останньому рядку, але присутній в масиві

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.2 3.2 4.3 4.4 4.5 5.1 3.7
1.4 5.2 5.3 6.4 6.4 6.7 6.7
1.4 7.2 7.3 7.4 8.1 8.1 8.3
1.4 8.1 9.2 9.3 9.4 9.5 10.4
1.6 10.2 10.2 10.2 10.3 11.1 11.5
1.8 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X: 11.2
Not founded in the first column and last row.
Process finished with exit code 0
```

Шуканий елемент займає весь стовпчик (окрім кута)


```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.1 3.2 4.3 4.4 4.5 5.1 3.7
1.1 5.2 5.3 6.4 6.4 6.7 6.7
1.1 7.2 7.3 7.4 8.1 8.1 8.3
1.1 8.1 9.2 9.3 9.4 9.5 10.4
1.1 10.2 10.2 10.2 10.3 11.1 11.5
1.1 11.2 12.3 12.4 12.4 12.5 13.6
1.8 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:1.1
The pos of X = [3][0]
Process finished with exit code 0
```

Шуканий елемент займає весь стовпчик

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.1 3.2 4.3 4.4 4.5 5.1 3.7
1.1 5.2 5.3 6.4 6.4 6.7 6.7
1.1 7.2 7.3 7.4 8.1 8.1 8.3
1.1 8.1 9.2 9.3 9.4 9.5 10.4
1.1 10.2 10.2 10.2 10.3 11.1 11.5
1.1 11.2 12.3 12.4 12.4 12.5 13.6
1.1 1.8 2.1 2.1 2.2 2.2 2.3
Enter X:1.1
The pos of X = [7][0]
Process finished with exit code 0
```

Шуканий елемент займає весь рядок(окрім кута)

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.1 3.2 4.3 4.4 4.5 5.1 3.7
1.1 5.2 5.3 6.4 6.4 6.7 6.7
1.1 7.2 7.3 7.4 8.1 8.1 8.3
1.1 8.1 9.2 9.3 9.4 9.5 10.4
1.1 10.2 10.2 10.2 10.3 11.1 11.5
1.1 11.2 12.3 12.4 12.4 12.5 13.6
1.1 2.1 2.1 2.1 2.1 2.1 2.1
Enter X:2.1
The pos of X = [7][3]
Process finished with exit code 0
```

Шуканий елемент займає весь рядок

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.1 3.2 4.3 4.4 4.5 5.1 3.7
1.1 5.2 5.3 6.4 6.4 6.7 6.7
1.1 7.2 7.3 7.4 8.1 8.1 8.3
1.1 8.1 9.2 9.3 9.4 9.5 10.4
1.1 10.2 10.2 10.2 10.3 11.1 11.5
1.1 11.2 12.3 12.4 12.4 12.5 13.6
2.1 2.1 2.1 2.1 2.1 2.1 2.1
Enter X:2.1
The pos of X = [7][0]
Process finished with exit code 0
```

Шуканий елемент займає весь рядок і стовпчик

```
1.1 1.2 1.3 2.4 2.5 2.7 3.7
1.1 3.2 4.3 4.4 4.5 5.1 3.7
1.1 5.2 5.3 6.4 6.4 6.7 6.7
1.1 7.2 7.3 7.4 8.1 8.1 8.3
1.1 8.1 9.2 9.3 9.4 9.5 10.4
1.1 10.2 10.2 10.2 10.3 11.1 11.5
1.1 11.2 12.3 12.4 12.4 12.5 13.6
1.1 1.1 1.1 1.1 1.1 1.1 1.1
Enter X:1.1
The pos of X = [7][0]
Process finished with exit code 0
```