

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»

Виконала:
Студентка групи ІМ-12
Миць Вікторія Ігорівна
Номер у списку групи: 19

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 19:

Задано матрицю дійсних чисел $A[m,n]$. Визначити присутність серед усіх елементів матриці будь-якого з чисел діапазону $[0,5]$ і його місцезнаходження (координати) методом двійкового пошуку (Алгоритм №1), якщо елементи кожного стовпчика окремо впорядковані за незбільшенням.

Код алгоритму на C:

1)

```
#include <stdio.h>

const int RANGE_UPPER = 5;
const int RANGE_LOWER = 0;

struct result {
    int row;
    int column;
} typedef result_t;

int is_in_range(double element) {
    return element >= RANGE_LOWER && element <= RANGE_UPPER;
}

void print_matrix(int m, int n, double matrix[m][n]) {
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            printf("%15lf ", matrix[i][j]);
        }
        printf("\n");
    }
}

void scan_matrix(int m, int n, double matrix[m][n]) {
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            scanf("%lf", &matrix[i][j]);
        }
    }
}

result_t find_value(int m, int n, double matrix[m][n]) {
    result_t result;
    result.row = -1;
    result.column = -1;

    for (int column_idx = 0; column_idx < n; column_idx++) {
        int begin = 0;
        int end = n - 1;

        while (begin <= end) {
            int mid = (begin + end) / 2;
            double element = matrix[mid][column_idx];

            if (is_in_range(element)) {
                result.row = mid;
                result.column = column_idx;
                break;
            } else if (element > RANGE_UPPER) {
                begin = mid + 1;
            } else {
                end = mid - 1;
            }
        }
    }
}
```

```

        end = mid - 1;
    }
}

    if (result.row >= 0 && result.column >= 0) {
        break;
    }
}

return result;
}

int main() {
    int m;
    scanf("%d", &m);

    int n;
    scanf("%d", &n);

    double matrix[m][n];
    scan_matrix(m, n, matrix);
    print_matrix(m, n, matrix);

    result_t result = find_value(m, n, matrix);

    if (result.row < 0 || result.column < 0) {
        printf("Haven't found any value in range [0,5]");
    } else {
        printf("Found a value in range [0,5]: %lf in row number %d and column number %d\n",
matrix[result.row][result.column], result.row + 1, result.column + 1);
    }

    return 0;
}

```

Результати тестувань

```
vika@MacBook-Pro-vika 2.1 Двійковий пошук % ./main < test_case_1.txt
10.700000      5.600000      100.330000     11.200000     56.400000      1.090000      8.990000      -3.400000
 5.100000      2.300000      100.010000     -10.500000    30.000000     -0.300000      3.000000     -50.000000
-0.300000      2.200000      4.000000      -30.000000    10.231000     -0.400000      2.000000     -50.300000
-50.300000     2.100000     -50.300000     -50.300000    -50.300000    -50.300000    -50.300000    -50.300000
-50.300000     2.020000     -50.300000     -50.300000    -50.300000    -50.300000    -50.300000    -50.300000
-50.350000     -50.360000    -50.340000     -50.300000    -50.300000    -50.399000    -50.300000    -50.390000
-50.350000     -50.360000    -50.340000     -50.300000    -50.300000    -50.399000    -50.300000    -50.390000

Found a value in range [0,5]: 2.100000 in row number 4 and column number 2

vika@MacBook-Pro-vika 2.1 Двійковий пошук % ./main < test_case_2.txt
10.700000      5.600000      100.330000      4.490000     56.400000      33.000000      8.990000
 5.100000      5.500000      100.010000      1.278000     30.000000      32.200000      3.000000
-0.300000      -3.000000      -9.000000        0.550000     10.231000      22.000000      2.000000
-12.400000     -12.400000     -12.400000     -12.400000    -12.400000     -12.400000     -12.400000
-14.500000     -14.600000     -14.700000     -14.800000    -14.900000     -14.990000     -14.999000
-146.554000    -15.400000     -16.110000     -16.300000    -23.421000     -17.700000     -30.300000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000

Found a value in range [0,5]: 1.278000 in row number 2 and column number 4

vika@MacBook-Pro-vika 2.1 Двійковий пошук % ./main < test_case_3.txt
10.700000      5.600000      100.330000      11.200000     56.400000      9.000000      8.990000      89.000000      50.000000
 5.100000      5.500000      100.010000     -10.500000    30.000000     -0.300000      6.400000      88.000000      40.000000
-0.300000      -3.000000      5.600000      -30.000000    10.231000     -0.400000      6.000000      88.000000      30.000000
-12.400000     -12.400000     -12.400000     -12.400000    -12.400000    -12.400000     -12.400000      87.000000      30.000000
-14.500000     -14.600000     -14.700000     -14.800000    -14.900000    -14.990000     -14.999000      87.000000      29.000000
-146.554000    -15.400000     -16.110000     -16.300000    -23.421000     -17.700000     -30.300000      87.000000      11.000000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000      3.300000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000      3.200000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000      3.200000

Found a value in range [0,5]: 3.300000 in row number 7 and column number 9

vika@MacBook-Pro-vika 2.1 Двійковий пошук % ./main < test_case_4.txt
10.700000      5.600000      100.330000      11.200000     56.400000      9.000000      8.990000      89.000000      50.000000
 5.100000      5.500000      100.010000     -10.500000    30.000000     -0.300000      6.400000      88.000000      40.000000
 4.255000      -3.000000      5.600000      -11.000000    10.231000     -0.400000      6.000000      88.000000      30.000000
 4.255000     -12.400000     -12.400000     -12.400000    -12.400000    -12.400000     -12.400000      87.000000      30.000000
 4.235000     -14.600000     -14.700000     -14.800000    -14.900000    -14.990000     -14.999000      87.000000      29.000000
-146.554000    -15.400000     -16.110000     -16.300000    -23.421000     -17.700000     -30.300000      87.000000      11.000000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000      3.300000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000      3.200000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000      3.200000

Found a value in range [0,5]: 4.235000 in row number 5 and column number 1

vika@MacBook-Pro-vika 2.1 Двійковий пошук % ./main < test_case_5.txt
10.700000      5.600000      100.330000      11.200000     56.400000      9.000000      8.990000      89.000000
 5.100000      5.500000      100.010000     -10.500000    30.000000     -0.300000      6.400000      88.000000
-3.000000      -3.000000      5.600000      -11.000000    10.231000     -0.400000      6.000000      88.000000
-4.000000     -12.400000     -12.400000     -12.400000    -12.400000    -12.400000     -12.400000      87.000000
-5.000000     -14.600000     -14.700000     -14.800000    -14.900000    -14.990000     -14.999000      87.000000
-146.554000    -15.400000     -16.110000     -16.300000    -23.421000     -17.700000     -30.300000      87.000000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000
-1566.554000   -154.000000    -17.000000     -16.500000    -23.900000     -50.000000     -37.000000     -3.000000

Haven't found any value in range [0,5]
```