**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 № 2.2**

з дисципліни  
«Алгоритми і структури даних»

Виконав Перевірила:

Студент групи ІП-03 Сергієнко А. А.  
Пашковський Євгеній Сергійович  
номер у списку групи: 18

Київ 2020

**Завдання**

1. Задано двовимірний масив (матрицю) цілих чисел ***A[m,n]*** або ***A[n,n]***, де ***m*** та ***n*** – натуральні числа (константи), що визначають розміри двовимірного масиву. Виконати сортування цього масиву або заданої за варіантом його частини у заданому порядку заданим алгоритмом (методом).

***Сортування повинно бути виконано безпосередньо у двовимірному масиві «на тому ж місці»***, тобто без перезаписування масиву та/або його будь-якої частини до інших одно- або двовимірних масивів, а також без використання спискових структур даних.

1. Розміри матриці **m** та **n** взяти самостійно у межах від 7 до 10.
2. При тестуванні програми необхідно підбирати такі вхідні набори початкових значень матриці, щоб можна було легко відстежити коректність виконання сортування і ця коректність була б протестована для всіх можливих випадків. З метою тестування дозволяється використовувати матриці меншого розміру.

Варіант 18:

Задано двовимірний масив (матрицю) цілих чисел ***A[m,n]***. Відсортувати окремо кожен стовпчик масиву методом швидкого сортування (методом Хоара) за незбільшенням.

**Текст програми**

#include <stdio.h>

int main () { // 0 1 2 3 4 5 6 7 8 9

float MATRIX\_WITH\_DATA[7][10] = { {5 , 5, 4 , 3 , 3, 3, 2.4, 1, 1, 1 /\*0\*/},

{5 , 5, 5 , 4 , 4, 4, 1 , 1, 1, 1 /\*1\*/},

{3.1, 3, 2 , 1 , 1, 1, 1 , 1, 1, 1 /\*2\*/},

{5 , 5, 4.5, 4.2, 4, 4, 1 , 1, 1, 0.3 /\*3\*/},

{5 , 5, 5 , 4.3, 4, 4, 1.8, 1, 1, 0.2 /\*4\*/},

{5 , 5, 5 , 4 , 4, 4, 2 , 1, 1, 0 /\*5\*/},

{5 , 5, 5 , 4 , 4, 4, 1 , 1, 1, 1 /\*6\*/}};

float NUBMER\_TO\_FIND = 0;

printf("Enter real number: ");

scanf("%f", &NUBMER\_TO\_FIND);

int ARRAY\_LENGTH = sizeof(MATRIX\_WITH\_DATA[0]) / sizeof(MATRIX\_WITH\_DATA[0][0]);

int i = 0, j = -1;

for (i; i < 7; i++) {// Проходимо по всіх рядках

int L = 0, R = ARRAY\_LENGTH - 1;

while (L < R) {

int MIDDLE\_INDEX = (L + R) / 2;

float middle = MATRIX\_WITH\_DATA[i][MIDDLE\_INDEX];

if (NUBMER\_TO\_FIND < middle) {

L = MIDDLE\_INDEX + 1;

} else {

R = MIDDLE\_INDEX;

}

}

if (R == L && MATRIX\_WITH\_DATA[i][R] == NUBMER\_TO\_FIND && MATRIX\_WITH\_DATA[i][L] == NUBMER\_TO\_FIND) {

j = L;

}

if (j != -1) {

break;

}

}

if (j != -1) {

printf("X: %d; Y: %d\n", i, j);

} else {

printf("There is no such number in the matrix!\n");

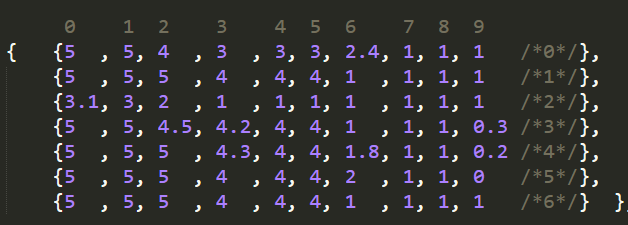
}

system("pause");

return 0;

}

**Вхідні дані**



**Тестування програми**

