**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»

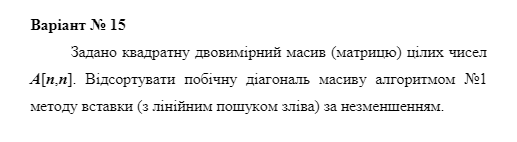
Виконала: Перевірила:

студентка групи ІП-05   
Лавринович Марія Юріївна  
номер у списку групи: 15

Київ 2020

**Постановка задачі**

Виконати сортування масиву або заданої за варіантом його частини у заданому порядку заданим алгоритмом (методом).



**Текст програми**

#include <stdio.h>

void print(int N, int M, int A[][M]) {

for (int R = 0; R < N; R++) {

for (int C = 0; C < M; C++)

printf("%d \t", A[R][C]);

printf("\n");

}

}

int sort(int numberRows, int numberCols, int A[][numberCols]) {

int rowIndex;

int currentSwapValue;

int currentSwapRow;

int currentSwapCol;

int temp;

for (int colIndex = numberCols - 1; colIndex >= 0; colIndex--) {

int rowIndex = numberCols - 1 - colIndex;

//printf("A[%d][%d] %d \n", rowIndex, colIndex, A[rowIndex][colIndex]);

currentSwapValue = A[rowIndex][colIndex]; //what we need to swap

currentSwapRow = rowIndex; //row of swap value

currentSwapCol = colIndex; //col of swap value

// printf("currentSwapRow: %d currentSwapCol: %d \n", currentSwapRow, currentSwapCol);

for (int swapColIndex = colIndex - 1; swapColIndex >= 0; swapColIndex--) {

//printf("currentInnerSwapRow: %d currentInnerSwapCol: %d \n", numberRows - 1 - swapColIndex, swapColIndex);

//printf("currentSwapValue: %d currentInnerSwapValue: %d \n", currentSwapValue, A[numberRows - 1 - swapColIndex][swapColIndex]);

if (currentSwapValue > A[numberRows - 1 - swapColIndex][swapColIndex]) {

currentSwapValue = A[numberRows - 1 - swapColIndex][swapColIndex];

currentSwapRow = numberRows - 1 - swapColIndex;

currentSwapCol = swapColIndex;

}

}

//swapping values

temp = A[rowIndex][colIndex];

A[rowIndex][colIndex] = A[currentSwapRow][currentSwapCol];

A[currentSwapRow][currentSwapCol] = temp;

//printf("A[%d][%d] %d \n", rowIndex, colIndex, A[rowIndex][colIndex]);

}

}

int main() {

int A[7][7] = {

{0, 0, 0, 0, 0, 0, 3},

{0, 0, 0, 0, 0, 8, 0},

{0, 0, 0, 0, -7, 0, 0},

{0, 0, 0, 11, 0, 0, 0},

{0, 0, 8, 0, 0, 0, 0},

{0, 4, 0, 0, 0, 0, 0},

{-5, 0, 0, 0, 0, 0, 0},

};

int num\_rows = sizeof(A) / sizeof(A[0]);

int num\_cols = sizeof(A[0]) / sizeof(A[0][0]);

printf("Number of rows %d \n", num\_rows);

printf("Number of cols %d \n", num\_cols);

print(num\_rows, num\_cols, A);

printf("\n");

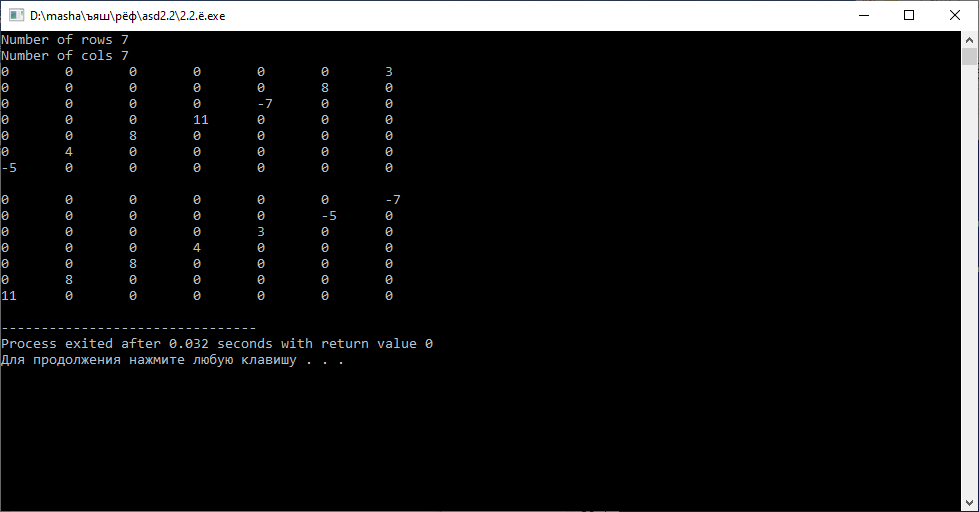
sort(num\_rows, num\_cols, A);

print(num\_rows, num\_cols, A);

return 0;

}

**Результати тестування програми**

****