

**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 №6
з дисципліни
«Алгоритми і структури даних»

Виконав:

студент групи ІМ-43
Черепов Олександр Павлович
номер у списку групи: 32

Перевірила:

Молчанова А. А.

Київ 2024

Постановка задачі

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.

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 сортування і ця коректність була б протестована для всіх можливих випадків. З метою тестування дозволяється використовувати матриці меншого розміру.

Варіант № 2

Задано двовимірний масив (матрицю) цілих чисел $A[m,n]$. Відсортувати окремо кожен рядок масиву алгоритмом №2 методу вставки (з лінійним пошуком справа) за незбільшенням.

Текст програми

```
#include <stdio.h>
int main(void) {
    int n, m, i, j, k, T;
    printf("Enter the size of your matrix (2 values separated by space): <rows collumns>\n");
    scanf("%d %d", &m, &n);

    if ((m<7 || m>10) || (n<7 || n>10)) {
        printf("Error! The size values should be between 7 and 10.");
        return 0;
    }
}
```

```

int matrix[m][n];

for (i=0; i<m; i++) {
    for (j=0; j<n; j++) {
        printf("Enter a value for element [%d][%d]: ", i, j);
        scanf("%d", &matrix[i][j]);
    }
}

printf("Input matrix:\n");
for (i=0; i<m; i++) {
    for (j=0; j<n; j++) {
        printf("%5d\t", matrix[i][j]);
    }
    printf("\n");
}

for (i=0; i<m; i++){
    for (j=1; j<n; j++) {
        T = matrix[i][j];
        k = j;

        while (T>matrix[i][k-1] && k>0) {
            matrix[i][k] = matrix[i][k-1];
            k--;
        }

        matrix[i][k] = T;
    }
}

printf("Sorted matrix:\n");
for (i=0; i<m; i++) {
    for (j=0; j<n; j++) {
        printf("%5d\t", matrix[i][j]);
    }
    printf("\n");
}

return 0;
}

```

Тестування програми

```

D:\education\EducationalWork\EducationalWork\ACД\000LABS\lab6\tests>test1.exe
Input matrix:
 5      6      4      2      8      10      11
 1      7      2      9      2      5      2
 8      3      5      1      8      3      2
11      11     11      9      8      7      6
15      12     10      7      4      1      1
29      30     31     32     33     34     35
22      23     24     25     26     27     28
Sorted matrix:
11      10      8      6      5      4      2
 9      7      5      2      2      2      1
 8      8      5      3      3      2      1
11      11     11      9      8      7      6
15      12     10      7      4      1      1
35      34     33     32     31     30     29
28      27     26     25     24     23     22

```

Висновок

На лабораторній роботі я розв'язував задачу по сортуванню рядків двовимірного масиву за незбільшенням, використовуючи Алгоритм№2, за допомогою мови програмування C. Для успішного виконання завдання я використовував Алгоритм№2, який обходить масив зліва направо та у випадку, якщо наступний елемент виявляється більшим за попередній, починається лінійний пошук комірки справа наліво для цього елемента та зсув всіх інших елементів вправо. На кожній ітерації циклу поточний елемент записувався в тимчасову комірку, тим самим звільняючи місце для інших. Відповідно, значення з цієї комірки порівнювалося з кожним елементом відсортованої частини масиву, рухаючись справа наліво, поки не було знайдено елемент, що є більшим за поточний. Одночасно з цим всі елементи до нього зсуваються, звільняючи місце, на яке стане елемент з тимчасової комірки.

У ході лабораторної роботи познайомися з труднощами сортування, а також з деякими алгоритмами сортування, зокрема Алгоритмом№2.

Отже, ця лабораторна робота була для мене надзвичайно важливою, адже її метою є навчитися сортуванню, яке є дуже корисним у вирішенні різних програмних задач.