

**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 та структури даних.**

Виконав:
студент групи ІМ-42
Федоренко Іван Русланович
номер варіанту: 30

Перевірила:
Молчанова А. А.

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).

Варіант № 30

Задано квадратну двовимірний масив (матрицю) цілих чисел $A[n,n]$. Відсортувати головну діагональ масиву методом швидкого сортування (методом Хоара) за незменшенням.

Текст програми:

```
#include <stdio.h>
```

```
void displayMatrix(int n, int A[n][n]) {
```

```
    for (int i = 0; i < n; i++) {
```

```
        for (int j = 0; j < n; j++) {
```

```
            printf("%3d", A[i][j]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
}
```

```
void quickSortDiagonalInPlace(int n, int A[n][n], int left, int right) {
```

```
    if (left >= right) return;
```

```
int pivot = A[left][left];
```

```
int i = left;
```

```
int j = right;
```

```
while (i <= j) {
```

```
    while (A[i][i] < pivot) i++;
```

```
    while (A[j][j] > pivot) j--;
```

```
    if (i <= j) {
```

```
        int temp = A[i][i];
```

```
        A[i][i] = A[j][j];
```

```
        A[j][j] = temp;
```

```
        i++;
```

```
        j--;
```

```
    }
```

```
}
```

```
quickSortDiagonalInPlace(n, A, left, j);
```

```
quickSortDiagonalInPlace(n, A, i, right);
```

```
}
```

```
int main() {  
  
    int n = 0;  
  
    printf("Enter rows and columns (n x n): ");  
  
    scanf("%d", &n);  
  
    int A[n][n];  
  
    printf("Enter elements:\n");  
    for (int i = 0; i < n; i++) {  
        for (int j = 0; j < n; j++) {  
            scanf("%d", &A[i][j]);  
        }  
    }  
  
    printf("\nBefore:\n");  
  
    displayMatrix(n, A);  
  
    quickSortDiagonalInPlace(n, A, 0, n - 1);  
  
    printf("\nAfter:\n");  
  
    displayMatrix(n, A);  
  
    return 0;  
}
```

```
}
```

Вивід програми:

Матриця 8на8, обернено відсортована.

```
D:\Documents\IVAN\Іван docs\КПІ\перший курс\АСД\lab 2.2>test
Enter rows and columns (n x n): 8
Enter elements:
  7   3   6   4   8   5   9 10
  6   7  10   5   3   7   4  8
  5   4   5   7   6   9   8  3
  8   9  15   4   8  12  10  2
  9  10   6   5   4  11  14  4
  4  11   8   6   7   2  16  6
  3   8   7   4   9   6   1 -7
  4  11   8   6   7   2  16  0

Before:
  7   3   6   4   8   5   9 10
  6   7  10   5   3   7   4  8
  5   4   5   7   6   9   8  3
  8   9  15   4   8  12  10  2
  9  10   6   5   4  11  14  4
  4  11   8   6   7   2  16  6
  3   8   7   4   9   6   1 -7
  4  11   8   6   7   2  16  0

After:
  0   3   6   4   8   5   9 10
  6   1  10   5   3   7   4  8
  5   4   2   7   6   9   8  3
  8   9  15   4   8  12  10  2
  9  10   6   5   4  11  14  4
  4  11   8   6   7   5  16  6
  3   8   7   4   9   6   7 -7
  4  11   8   6   7   2  16  7

D:\Documents\IVAN\Іван docs\КПІ\перший курс\АСД\lab 2.2>
```

Матриця 7на7, вже відсортована діагональ:

Enter rows and columns (n x n): 7

Enter elements:

1	4	8	7	12	14	5
5	2	6	3	9	11	8
7	4	3	15	18	6	9
6	9	10	4	13	17	12
8	3	5	14	5	7	10
13	11	9	6	12	6	4
11	14	16	17	8	9	7

Before:

1	4	8	7	12	14	5
5	2	6	3	9	11	8
7	4	3	15	18	6	9
6	9	10	4	13	17	12
8	3	5	14	5	7	10
13	11	9	6	12	6	4
11	14	16	17	8	9	7

After:

1	4	8	7	12	14	5
5	2	6	3	9	11	8
7	4	3	15	18	6	9
6	9	10	4	13	17	12
8	3	5	14	5	7	10
13	11	9	6	12	6	4
11	14	16	17	8	9	7

Матриця 7на7, **не відсортована** елементів головної діагоналі:

Enter rows and columns (n x n): 7

Enter elements:

-1	4	8	7	12	14	5
5	-2	6	3	9	11	8
7	4	3	15	18	6	9
6	9	10	-4	13	17	12
8	3	5	14	15	7	10
13	11	9	6	12	6	4
11	14	16	17	8	9	0

Before:

-1	4	8	7	12	14	5
5	-2	6	3	9	11	8
7	4	3	15	18	6	9
6	9	10	-4	13	17	12
8	3	5	14	15	7	10
13	11	9	6	12	6	4
11	14	16	17	8	9	0

After:

-4	4	8	7	12	14	5
5	-2	6	3	9	11	8
7	4	-1	15	18	6	9
6	9	10	0	13	17	12
8	3	5	14	3	7	10
13	11	9	6	12	6	4
11	14	16	17	8	9	15

Матриця 10на10, не відсортована.

```

D:\Documents\IVAN\Іван docs\КПІ\перший курс\АСД\lab 2.2>test
Enter rows and columns (n x n): 10
Enter elements:
14  4  6 10 15 12 18  7 11  8
 3 11 19 17 10  5  6  4 13  9
 9  6 14 14  7  8 12 11  3 16
 8 15  7  8  4 10  9 12  5 14
19 18 12 13  9  6  4 15 11  3
17  9  5 16 11 13  7 10  4  6
15 14  8  7 12  9  3  5  6 13
 5  8 16 14  6 10 11 18 12 19
 3  4 10  6 14  8  9  7  7 11
11 12  9 15  3  5 13 19  8  8

Before:
14  4  6 10 15 12 18  7 11  8
 3 11 19 17 10  5  6  4 13  9
 9  6 14 14  7  8 12 11  3 16
 8 15  7  8  4 10  9 12  5 14
19 18 12 13  9  6  4 15 11  3
17  9  5 16 11 13  7 10  4  6
15 14  8  7 12  9  3  5  6 13
 5  8 16 14  6 10 11 18 12 19
 3  4 10  6 14  8  9  7  7 11
11 12  9 15  3  5 13 19  8  8

After:
 3  4  6 10 15 12 18  7 11  8
 3  7 19 17 10  5  6  4 13  9
 9  6  8 14  7  8 12 11  3 16
 8 15  7  8  4 10  9 12  5 14
19 18 12 13  9  6  4 15 11  3
17  9  5 16 11 11  7 10  4  6
15 14  8  7 12  9 13  5  6 13
 5  8 16 14  6 10 11 14 12 19
 3  4 10  6 14  8  9  7 14 11
11 12  9 15  3  5 13 19  8 18

D:\Documents\IVAN\Іван docs\КПІ\перший курс\АСД\lab 2.2>|

```

Матриця 10на10, обернено відсортована.

Enter rows and columns (n x n): 10

Enter elements:

10	3	5	7	9	8	6	4	11	12
8	10	6	15	14	7	3	12	4	10
11	5	8	7	6	4	9	14	15	13
7	12	10	7	3	5	4	8	6	14
4	9	8	6	5	15	7	13	3	5
6	7	4	5	8	5	3	9	15	7
11	8	10	3	12	13	4	5	7	9
9	5	7	4	10	11	6	12	8	14
15	12	6	5	4	8	9	14	10	3
13	6	3	11	8	15	4	10	9	1

Before:

10	3	5	7	9	8	6	4	11	12
8	10	6	15	14	7	3	12	4	10
11	5	8	7	6	4	9	14	15	13
7	12	10	7	3	5	4	8	6	14
4	9	8	6	5	15	7	13	3	5
6	7	4	5	8	5	3	9	15	7
11	8	10	3	12	13	4	5	7	9
9	5	7	4	10	11	6	12	8	14
15	12	6	5	4	8	9	14	10	3
13	6	3	11	8	15	4	10	9	1

After:

1	3	5	7	9	8	6	4	11	12
8	4	6	15	14	7	3	12	4	10
11	5	5	7	6	4	9	14	15	13
7	12	10	5	3	5	4	8	6	14
4	9	8	6	7	15	7	13	3	5
6	7	4	5	8	8	3	9	15	7
11	8	10	3	12	13	10	5	7	9
9	5	7	4	10	11	6	10	8	14
15	12	6	5	4	8	9	14	10	3
13	6	3	11	8	15	4	10	9	12

D:\Documents\IVAN\Іван docs\КПІ\перший курс\АСД\lab 2.2>

Висновок:

Я закріпив знання алгоритму Хоара, сортуючи головну діагональ квадратної матриці за незменшенням.