

Міністерство освіти і науки України
Національний технічний університет України «Київський
політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки
 Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 6 з
дисципліни «Алгоритми та структури
даних-1. Основи алгоритмізації»

«Дослідження лінійного пошуку в
послідовностях»

Варіант 10

Виконав студент IП-13 Дейнега Владислав Миколайович
(шифр, прізвище, ім'я, по батькові)

Перевірив Вечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Київ 2021

Лабораторна робота 8

Дослідження алгоритмів пошуку та сортування

Мета – дослідити алгоритми пошуку та сортування, набути практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

Варіант 10

Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опису дійсного двовимірного масиву розміром 4 x 8.
2. Ініціювання масиву із мінімальних значень стовпців двовимірного масиву.
3. Відсортувати методом Шела за спаданням.

Постановка задачі

Результатом є відсортована за спаданням одновимірний масив, який складається із мінімальних елементів стовпців двовимірного масиву.

Побудова математичної моделі

Таблиця імен змінних

Змінна	Тип	Ім'я	Призначення
Матриця	Дійсний	matr	Проміжне дане
Масив мінімальних елементів	Дійсний	arr_min	Результат
Кількість рядків	Цілий	rows	Проміжне дане
Кількість стовпців	Цілий	cols	Проміжне дане

Функція matr_fill

Змінна	Тип	Ім'я	Призначення
Матриця	Дійсний	matr	Проміжне дане
Кількість рядків	Цілий	rows	Проміжне дане
Кількість стовпців	Цілий	cols	Проміжне дане
Лічильник i	Цілий	i	Проміжне дане

Лічильник j	Цілий	j	Проміжне дане
-------------	-------	---	---------------

Функція arr_min_inic

Змінна	Тип	Ім'я	Призначення
Матриця	Дійсний	matr	Проміжне дане
Масив мінімальних елементів	Дійсний	arr_min	Результат
Кількість рядків	Цілий	rows	Проміжне дане
Кількість стовпців	Цілий	cols	Проміжне дане
Проміжна змінна	Цілий	min	Проміжне дане
Лічильник i	Цілий	i	Проміжне дане
Лічильник j	Цілий	j	Проміжне дане

Функція arr_out

Змінна	Тип	Ім'я	Призначення
Масив мінімальних елементів	Дійсний	arr	Результат
Кількість елементів	Дійсний	elem	Проміжне дане
Лічильник i	Цілий	i	Проміжне дане

Функція matr_out

Змінна	Тип	Ім'я	Призначення
Матриця	Дійсний	matr	Проміжне дане
Кількість рядків	Цілий	rows	Проміжне дане
Кількість стовпців	Цілий	cols	Проміжне дане
Лічильник i	Цілий	i	Проміжне дане
Лічильник j	Цілий	j	Проміжне дане

Функція arr_sort

Змінна	Тип	Ім'я	Призначення
Масив мінімальних елементів	Дійсний	arr	Результат
Кількість рядків	Цілий	rows	Проміжне дане
Проміжна змінна	Дійсний	temp	Проміжне дане
Кількість кроків	Цілий	step	Проміжне дане
Лічильник i	Цілий	i	Проміжне дане
Лічильник j	Цілий	j	Проміжне дане

Математичне формулювання задачі зводиться до сортування масива способом Шелу, елементи якого є мінімальними елементами стовпців матриці.

Псевдокод

Початок

Введення rows, cols

matr_fill(matr, rows, cols)

matr_out(rows, cols, matr)

arr_min_inic(matr, arr_min, rows, cols)

arr_out(arr_min, cols)

arr_sort(arr_min, cols)

arr_out(arr_min, cols)

Кінець

Функція matr_fill(matr, rows, cols)

для i від 0 до rows повторити

для j від 0 до cols повторити

matr[i][j] = rand(0, 20)

все повторити

все повторити

все функція

Функція matr_out(rows, cols, matr)

для i від 0 до rows повторити

для j від 0 до cols повторити

Вивести matr[i][j]

все повторити

все повторити

все функція

Функція arr_min_inic(matr, arr_min, rows, cols)

для i від 0 до cols повторити

min = matr[0][i]

для j від 0 до rows повторити

Якщо matr[j][i] < min

min = matr[j][i]

Все якщо

все повторити

arr_min[i] = min

все повторити

все функція

Функція arr_out(arr, elem)

для i від 0 до elem повторити

Вивести arr[i]

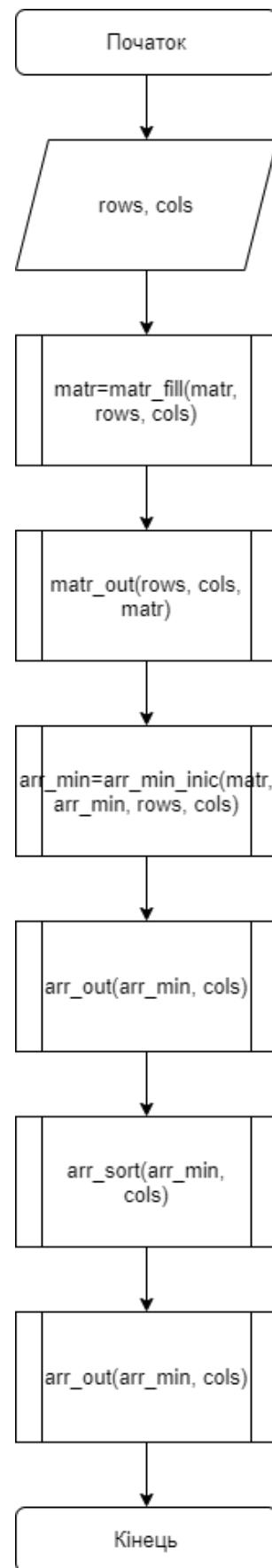
все повторити

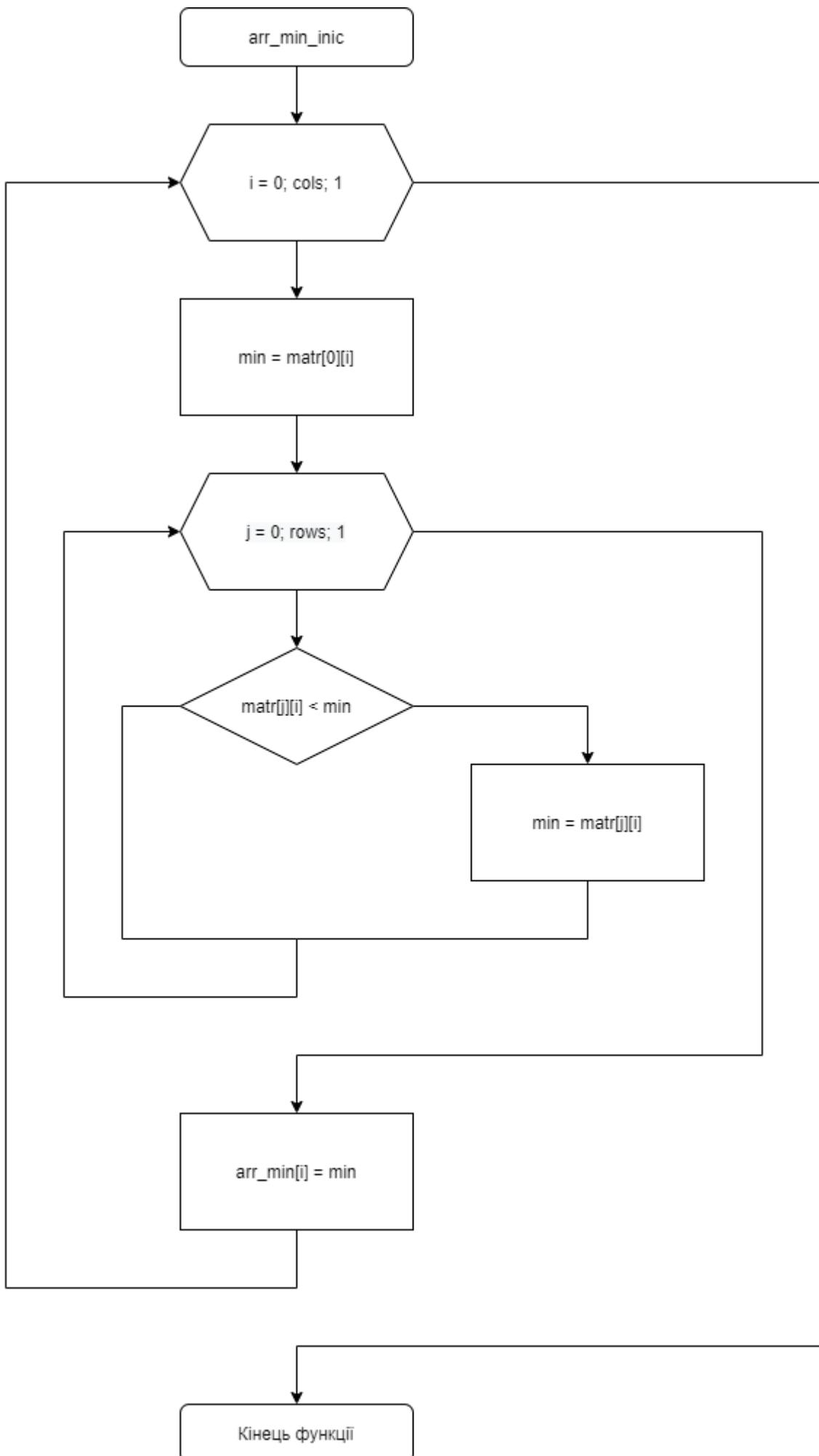
все функція

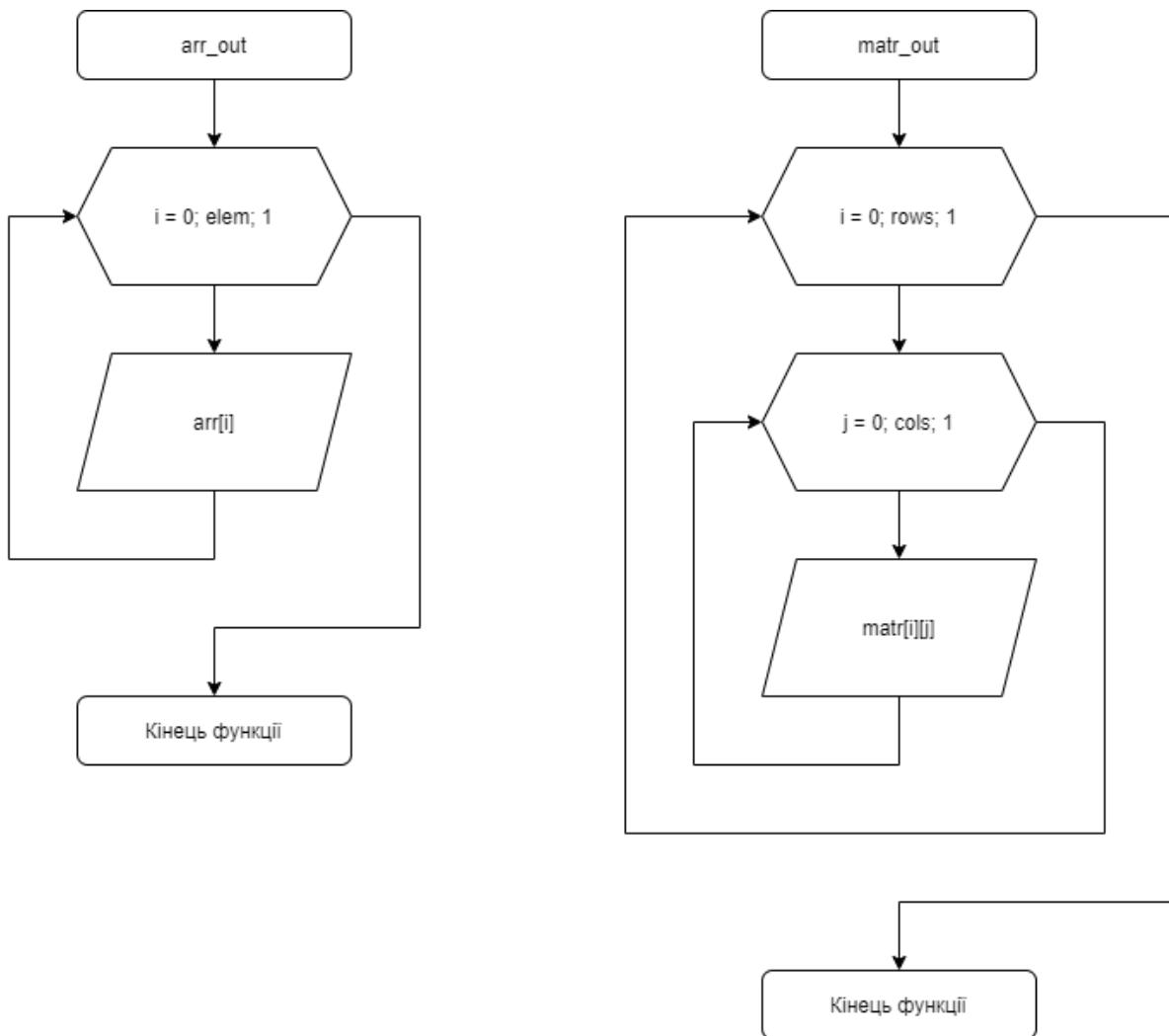
Функція arr_sort(arr, rows)

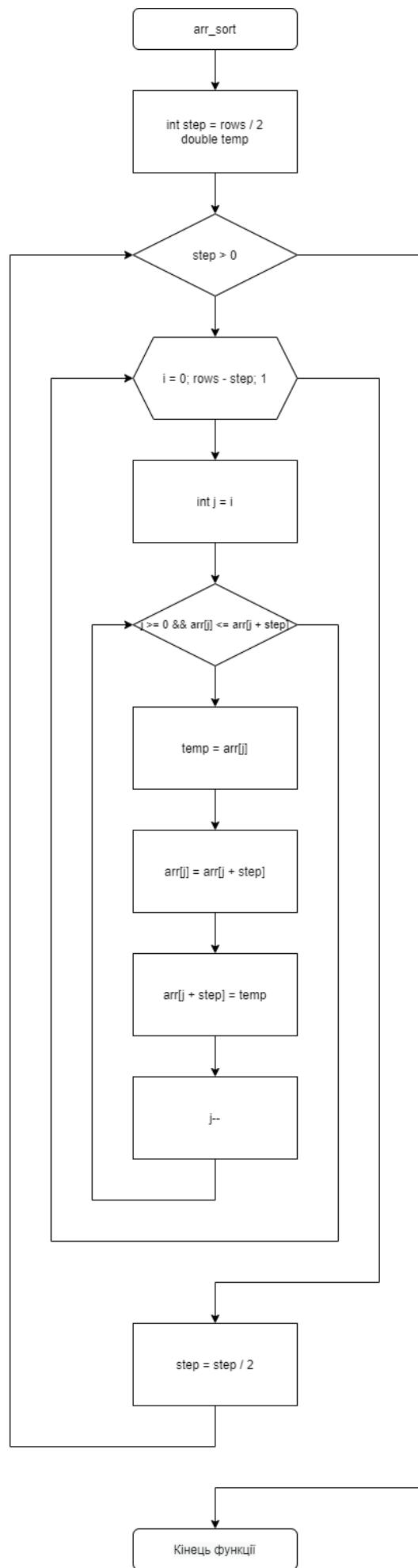
```
step = rows / 2
поки step > 0
    для i від 0 до rows - step повторити
        j = i
        поки j >= 0 && arr[j] <= arr[j + step]
            temp = arr[j]
            arr[j] = arr[j + step]
            arr[j + step] = temp
            j--
        повторити
    все повторити
    step = step / 2
повторити
все функція
```

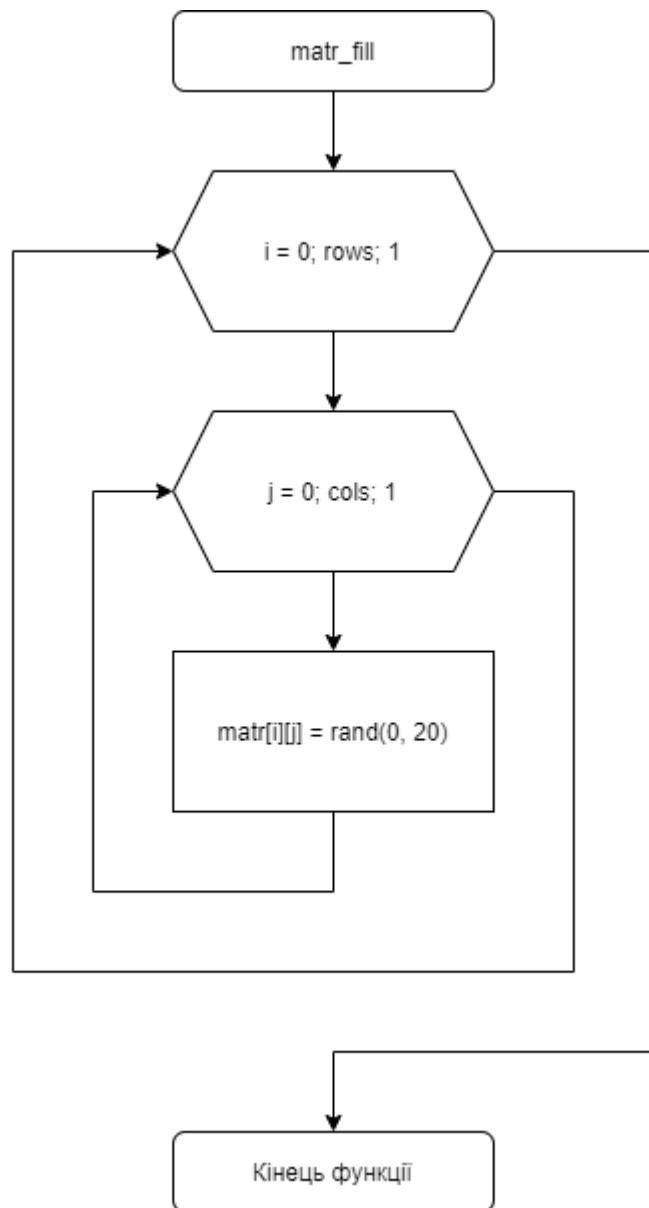
Блок-схема











Код програми

```
1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4
5  using namespace std;
6
7
8  void arr_out(float*, int);
9  void matr_out(int, int, float**);
10 void matr_fill(float**, int, int);
11 void arr_min_inic(float**, float*, int, int);
12 void matr_delet(int, float**);
13 void arr_delet(int, float*);
14 void arr_sort(float*, int);
15
16 int main()
17 {
18     srand(time(NULL));
19     int rows, cols;
20     cout << "Enter number of rows" << endl;
21     cin >> rows;
22     cout << "Enter number of cols" << endl;
23     cin >> cols;
24
25     float* arr_min = new float[cols];
26
27     float** matr = new float* [rows];
28     for (int i = 0; i < rows; i++)
29     {
30         matr[i] = new float[cols];
31     }
32
33     matr_fill(matr, rows, cols);
34     cout << "Matrix" << endl;
35     matr_out(rows, cols, matr);
36     arr_min_inic(matr, arr_min, rows, cols);
37     cout << "Array minimum" << endl;
38     arr_out(arr_min, cols);
39
40     arr_sort(arr_min, cols);
41
42     cout << "Sorted array minimum" << endl;
43     arr_out(arr_min, cols);
44
45     matr_delet(rows, matr);
46     arr_delet(rows, arr_min);
47 }
48
49 void arr_min_inic(float** matr, float* arr_min, int rows, int cols)
50 {
51     for (int i = 0; i < cols; i++)
52     {
53         float min = matr[0][i];
54         for (int j = 0; j < rows; j++)
55         {
56             if (matr[j][i] < min)
57             {
58                 min = matr[j][i];
59             }
60         }
61         arr_min[i] = min;
62     }
63 }
64
65 void matr_delet(int rows, float** matr)
66 {
67     for (int i = 0; i < rows; i++)
68     {
69         delete[] matr[i];
70     }
71     delete[] matr;
72 }
```

```
74     void arr_out(float* arr, int elem)
75     {
76         for (int i = 0; i < elem; i++)
77         {
78             cout << arr[i] << " ";
79         }
80         cout << endl;
81     }
82
83     void matr_fill(float** matr, int rows, int cols)
84     {
85         for (int i = 0; i < rows; i++)
86         {
87             for (int j = 0; j < cols; j++)
88             {
89                 matr[i][j] = rand() % 21;
90             }
91         }
92     }
93
94     void matr_out(int rows, int cols, float** matr)
95     {
96         for (int i = 0; i < rows; i++)
97         {
98             for (int j = 0; j < cols; j++)
99             {
100                 cout << setw(2) << matr[i][j] << "\t";
101             }
102             cout << endl;
103         }
104     }
105
106    void arr_delete(int elem, float* arr)
107    {
108        delete[] arr;
109    }
```

```
111    void arr_sort(float* arr, int rows)
112    {
113        int step = rows / 2;
114        double temp;
115        while (step > 0)
116        {
117            for (int i = 0; i < rows - step; i++)
118            {
119                int j = i;
120                while (j >= 0 && arr[j] <= arr[j + step])
121                {
122                    temp = arr[j];
123                    arr[j] = arr[j + step];
124                    arr[j + step] = temp;
125                    j--;
126                }
127            }
128            step = step / 2;
129        }
130    }
```

```
Microsoft Visual Studio Debug Console
Enter number of rows
4
Enter number of cols
8
Matrix
16      14      12      19      2      13      2      5
15      12      19      1       2      13      9      10
17      20      0       11      17      3       0       1
3       4       8       12      2       17      19      12
Array minimum
3 4 0 1 2 3 0 1
Sorted array minimum
4 3 3 2 1 1 0 0
```

```
Microsoft Visual Studio Debug Console
Enter number of rows
4
Enter number of cols
8
Matrix
8       16      15      15      8       1       6       11
0       11      13      20      9       11      11      2
15      5       12      17      18      9       9       13
4       13      20      11      7       2       17      14
Array minimum
0 5 12 11 7 1 6 2
Sorted array minimum
12 11 7 6 5 2 1 0
```

```
Microsoft Visual Studio Debug Console
Enter number of rows
4
Enter number of cols
8
Matrix
7       14      8       4       6       14      4       15
10      2       0       14      7       5       5       3
14      2       11      20      3       14      13      6
11      17      16      9       6       12      20      12
Array minimum
7 2 0 4 3 5 4 3
Sorted array minimum
7 5 4 4 3 3 2 0
```

Висновок

Під час виконання лабораторної роботи, я дослідив особливості алгоритмів пошуку та сортування та набув практичних навичок їх використання.