

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное образовательное
учреждение высшего образования
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ(МИИТ))**

Кафедра «Вычислительные системы, сети и информационная безопасность»

**ОТЧЕТ ПО ДИСЦИПЛИНЕ
«АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ»
ЛАБОРАТОРНАЯ РАБОТА №7**

Направление: 10.03.01 Информационная безопасность

Профиль: Безопасность компьютерных систем

Выполнил:
студент группы УИБ-115
Клепиков Степан Даниилович

Проверил:

(должность, ФИО)

(должность, ФИО)

Москва 2021 г.

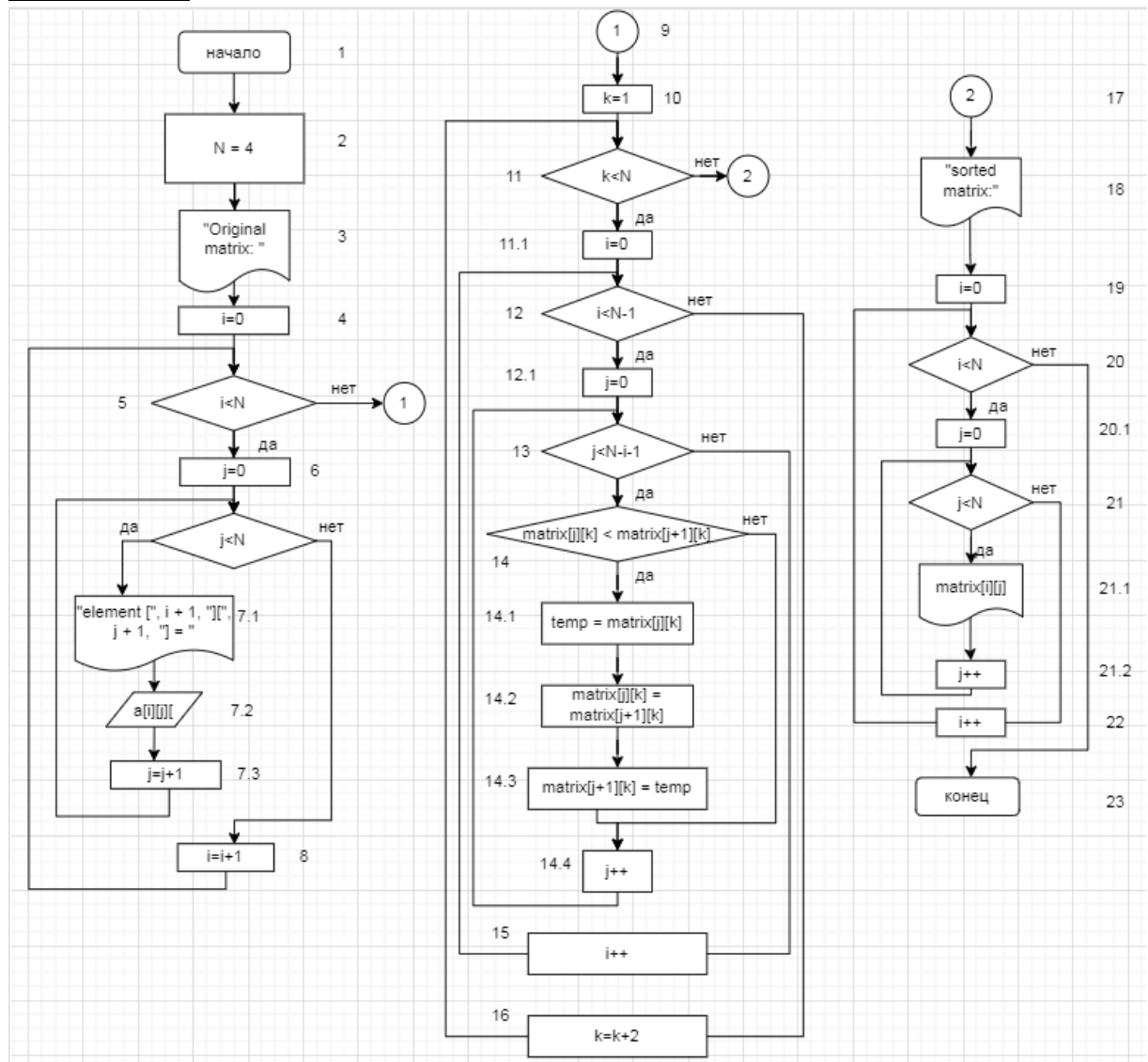
Задание

Задана матрица целых чисел. Выполнить сортировку элементов по убыванию в чётных столбцах заданной матрицы.

1. Таблица имён:

Исходные данные		
N	Целочисленное	Размерность матрицы
Matrix[N][N]	Целочисленное	-
Рабочие переменные		
temp	Целочисленное	Временная переменная
i	Целочисленное	Счетчик (строка)
j	Целочисленное	Счетчик (столбец)
k	Целочисленное	Счетчик (четные столбцы)
Результат		
-	-	-

2. Блок-схема:



3. Отладочный пример:

1. Начало
2. Присвоение переменной N значения 4
3. Вывод "Original matrix "
4. Присвоение переменной i значения 0
5. Проверка условия $(i < N) \ i=0, N=4 \Rightarrow$ истинно
6. Присвоение переменной j значения 0
7. Проверка условия $(j < N) \ j=0, N=4 \Rightarrow$ истинно
 - 7.1. вывод "Enter element" [0][0]
 - 7.2. присвоение [0][0] элементу матрицы значения **(16)**
 - 7.3. присвоение переменной j значения j+1
7. Проверка условия Проверка условия $(j < N) \ j=1, N=4 \Rightarrow$ истинно
 - 7.1 вывод "Enter element" [0][1]
 - 7.2 присвоение [0][1] элементу матрицы значения **(15)**
 - 7.3 присвоение переменной j значения j+1
-
7. Проверка условия $(j < N) \ j=4, N=4 \Rightarrow$ ложно
8. Присвоение переменной i значения i+1
5. Проверка условия $(i < N) \ i=1, N=4 \Rightarrow$ истинно
6. Присвоение переменной j значения 0
-
7. Проверка условия $(j < N) \ j=4, N=4 \Rightarrow$ ложно
8. Присвоение переменной i значения i+1
5. Проверка условия $(i < N) \ i=4, N=4 \Rightarrow$ ложно
8. Присвоение переменной j значения 0
9. Выход из цикла ввода матрицы, переход к циклу ее обработки
10. Присвоение переменной k значения 1
11. Проверка условия $(k < N) \ k=1, N=4 \Rightarrow$ истинно
 - 11.1 Присвоение переменной i значения 0
12. Проверка условия $(i < N-1) \ i=0, N-1=3 \Rightarrow$ истинно
 - 12.1 Присвоение переменной j значения 0
13. Проверка условия $(j < N-i-1) \ j=0, i=0, N-0-1=3 \Rightarrow$ истинно
14. Проверка условия $(\text{matrix}[j][k] < \text{matrix}[j+1][k]) \ (15 > 11 \text{ истинно})$
 - 14.1 присвоение переменной temp значения $\text{matrix}[j][k] \ (15)$
 - 14.2 присвоение переменной $\text{matrix}[j][k]$ значения $\text{matrix}[j+1][k]$
(15 -> 11)
 - 14.3 присвоение переменной $\text{matrix}[j+1][k]$ значения temp (11 -> 15)
 - 14.4 присвоение переменной j значения j+1
-
15. присвоение переменной i значения i+1
16. присвоение переменной k значения k+2
17. Выход из цикла обработки матрицы, переход к циклу вывода

18.Вывод "Sorted matrix: "

19.присвоение переменной i значения 0

20.Проверка условия (i<N) i=0, N=4 => истинно

20.1 присвоение переменной j значения 0

21.Проверка условия (j<N) j=0, N=4 => истинно

21.1 вывод [0][0] элемента

21.2 присвоение переменной j значения j+1

.....

21. Проверка условия (j<N) j=4, N=4 => ложно

22.присвоение переменной i значения i+1

20.Проверка условия (i<N) i=1, N=4 => истинно

20.1 присвоение переменной j значения 0

.....

20.Проверка условия (i<N) i=1, N=4 => ложно

23.конец

4. Код программы:

```
#include <iostream>//подключение библиотеки ввода-вывода
```

```
#include <string>//подключение библиотеки для работы со строками
```

```
using namespace std; //подключение пространства имён std
```

```
int main()//точка входа в программу
```

```
{
```

```
    const int N = 4; // объявление константы типа integer
```

```
    int i, j, sum; // объявление переменных типа integer
```

```
    int sum_temp = 0;
```

```
    int matrix[N][N];
```

```
    cout << "Original matrix: " << endl;
```

```
    for (i = 0; i < N; i++)// цикл ввода матрицы
```

```
    {
```

```
        for (j = 0; j < N; j++)// вложенный цикл ввода матрицы
```

```
        {
```

```
            cout << "element [" << i + 1 << "][" << j + 1 << "] = ";
```

```
            cin << matrix[i][j] ;// ввод элемента матрицы
```

```
        }
```

```
        cout << endl ;
```

```
    }
```

```
    for (k = 1; k < N; k += 2) { // цикл для четных столбцов матрицы
```

```
        for (i = 0; i < N - 1; i++) { // цикл ввода матрицы
```

```

        for (j = 0; j < N - i - 1; j++) { //вложенный цикл ввода матрицы
            if (matrix[j][k] < matrix[j + 1][k]) { // сортировка элементов
                temp = matrix[j][k]; // меняем элементы местами
                matrix[j][k] = matrix[j + 1][k];
                matrix[j + 1][k] = temp;
            }
        }
    }
}
cout << endl << "Sorted Matrix: " << endl;
for (i = 0; i < N; i++) // цикл вывода матрицы
{
    for (j = 0; j < N; j++) // вложенный цикл вывода матрицы
    {
        cout << matrix[i][j] << " "; // вывод элемента матрицы
    }
    cout << endl;
}
}

return 0; //успешное завершение программы
}

```

5. Результат выполнения работы программы:

```

element [1][1] = 16
element [1][2] = 15
element [1][3] = 14
element [1][4] = 13

element [2][1] = 12
element [2][2] = 11
element [2][3] = 10
element [2][4] = 9

element [3][1] = 8
element [3][2] = 7
element [3][3] = 6
element [3][4] = 5

element [4][1] = 4
element [4][2] = 3
element [4][3] = 2
element [4][4] = 1

Sorted Matrix:
16 3 14 1
12 7 10 5
8 11 6 9
4 15 2 13

```

6. Вывод:

В ходе выполнения работы были изучены базовые алгоритмы языка C++, библиотека <algorithm>.

Был проделан анализ работы, на основании которого была составлена блок-схема.

На контрольных примерах мы убедились, что программа работает корректно.

Был оформлен комплект документации на программный код.