

**Московский государственный технический  
университет им. Н.Э. Баумана.**

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Основы информатики»

Отчет по лабораторной работе № 6

«Сортировка одномерного числового массива»

Выполнил:

студент группы ИУ5-13

Терентьев Владислав

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Козлов А. Д.

Подпись и дата:

Москва, 2018 г.

## 1. Постановка задачи

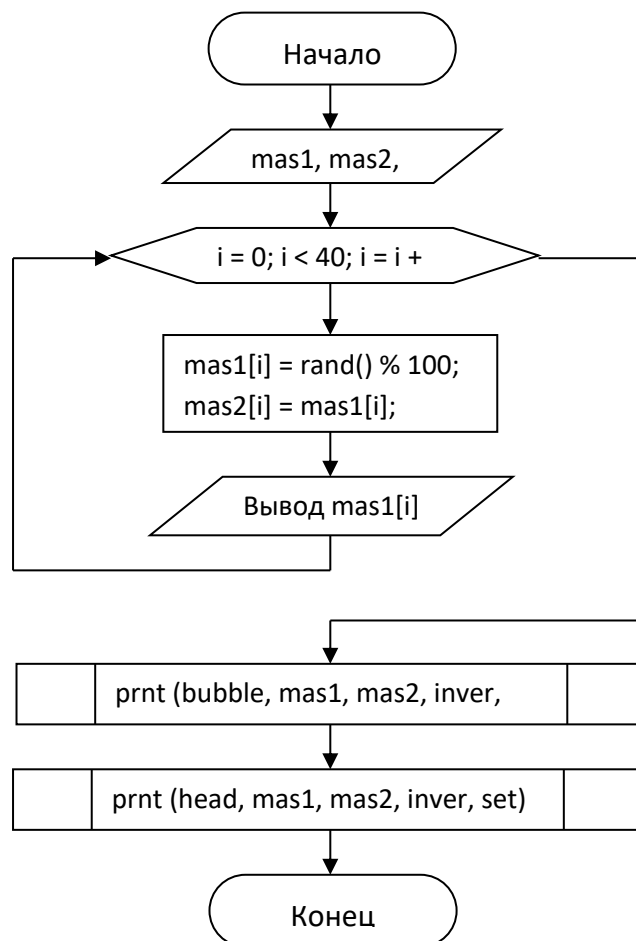
Написать программу сортировки, предварительно заполненного случайными числами, массива методами пузырька и выбора.

## 2. Разработка алгоритма

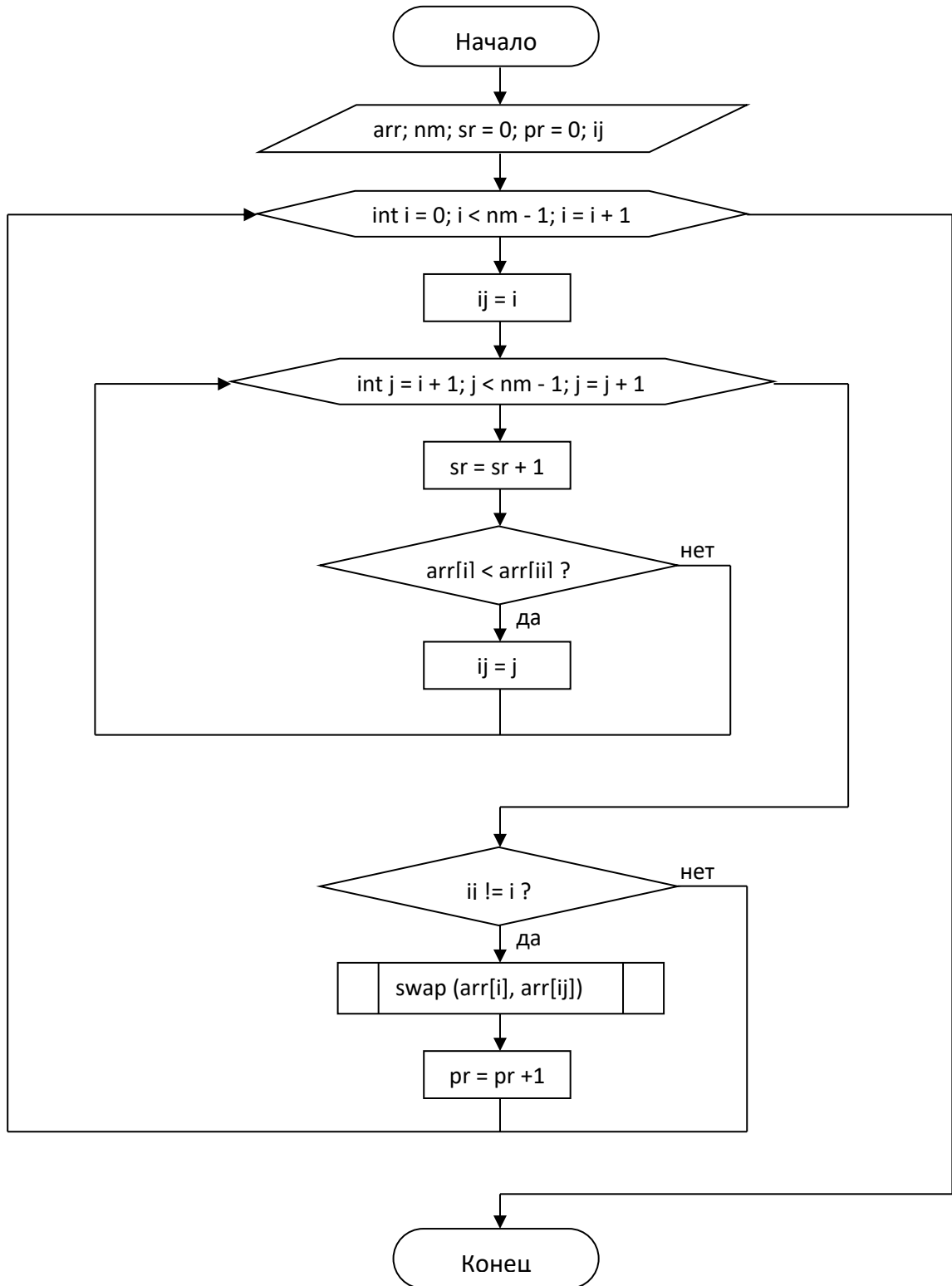
**Описание переменных:** переменные типа `int`: `mas1[40]` – массив из 40-ка целочисленных элементов, который нужно будет отсортировать; `mas2[40]` – для копии массива `mas1`; `i, j` – для цикла (индекс элемента массива); `sr` – счетчик сравнений; `pr` – счетчик перестановок; `ij` – индекс элемента массива; `nmb` – номер списка вывода.

**Описание функций:** функции типа `void`: `bubble` – 2 входных параметра (целочисленный массив и целое число), сортирует пузырьком определенное число элементов массива; `head` – 2 входных параметра (целочисленный массив и целое число), сортирует выбором определенное число элементов массива; `inver` – 2 входных параметра (целочисленный массив и целое число), меняет знак определенного числа элементов массива; `set` – 3 входных параметра (два целочисленных массива и целое число), присваивает определенное число элементов второго массива соответствующим элементам первого массива; `prnt` – 5 входных параметров (функция типа `void` (метод сортировки), 2 целочисленных массива, две функции типа `void` (`inver` и `set`), сортирует неупорядоченный/упорядоченный/с инверсией знаков массив из 40/20/10 элементов и выводит результат.

Укрупненная схема алгоритма:



### Схема блока функции сортировки выбором “head”:



### 3. Текст программы

```
#include "pch.h"
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <iomanip>

using namespace std;
```

```

void bubble(int *, int);
void head(int *, int);
void inver(int *, int);
void set(int *, int *, int);
void prnt(void(*)(int *, int), int *, int *, void(*)(int *, int), void(*)(int *, int *,
int));

int main()
{
    setlocale(LC_ALL, "Russian");
    int mas1[40], //для случайных чисел (исходный)
        mas2[40], //для цикла (индекс элемента массива)
        i; //для цикла (индекс элемента массива)
    srand(time(0));
    cout << "Массив случайных чисел: " << endl;
    for (i = 0; i < 40; i++) {
        mas1[i] = rand() % 100;
        mas2[i] = mas1[i];
        cout << mas1[i] << " ";
    }
    cout << endl << setfill(char('-')) << setw(70) << "-" << endl << endl << endl <<
"Сортировка пузырьком:" << endl << endl;
    prnt(bubble, mas1, mas2, inver, set);
    cout << setfill(char('-')) << setw(70) << "-" << endl << endl << endl <<
"Сортировка выбором:" << endl << endl;
    prnt(head, mas1, mas2, inver, set);
    system("pause");
    return 0;
}

void bubble(int *arr, int nm) {
    int sr = 0, //счетчик сравнений
        pr = 0; //счетчик перестановок
    for (int i = 0; i < nm - 1; i++) {
        for (int j = 0; j < nm - i - 1; j++) {
            sr++;
            if (arr[j] > arr[j + 1]) {
                swap(arr[j], arr[j + 1]);
                pr++;
            }
        }
        if (pr == 0) break;
    }
    cout << "Количество сравнений: " << sr << endl << "Количество перестановок: " <<
pr << endl << "Отсортированный массив: " << endl;
    for (int i = 0; i < nm; i++) {
        cout << arr[i] << " ";
    }
    cout << endl << endl;
}

void head(int *arr, int nm) {
    int sr = 0, //счетчик сравнений
        pr = 0, //счетчик перестановок
        ij; //индекс элемента массива
    for (int i = 0; i < nm - 1; i++) {

```

```

        ij = i;
        for (int j = i + 1; j < nm; j++) {
            sr++;
            if (arr[j] < arr[ij]) {
                ij = j;
            }
        }
        if (ij != i) {
            swap(arr[i], arr[ij]);
            pr++;
        }
    }
    cout << "Количество сравнений: " << sr << endl << "Количество перестановок: " <<
pr << endl << "Отсортированный массив: " << endl;
    for (int i = 0; i < nm; i++) {
        cout << arr[i] << " ";
    }
    cout << endl << endl;
}

void inver(int *arr, int nm) {
    for (int i = 0; i < nm; i++) {
        arr[i] = -arr[i];
    }
}

void set(int *arr1, int *arr2, int nm) {
    for (int i = 0; i < nm; i++) {
        arr1[i] = arr2[i];
    }
}

void prnt(void(*sort)(int *, int), int *mas1, int *mas2, void(*inver)(int *, int),
void(*set)(int *, int *, int)) {
    int nmb = 0; //номер списка вывода
    for (int i = 1, k = 40; i < 4; i++, k = k / 2) {
        cout << ++nmb << ". Сортировка неупорядоченного массива из " << k << "
элементов:" << endl;
        sort(mas2, k);
        cout << ++nmb << ". Сортировка упорядоченного массива из " << k << "
элементов:" << endl;
        sort(mas2, k);
        inver(mas2, k);
        cout << ++nmb << ". Сортировка упорядоченного массива из " << k << "
элементов с инверсией знаков значений элементов:" << endl;
        sort(mas2, k);
        set(mas2, mas1, k);
    }
}

```

## 4. Анализ результатов

Скриншот программы: <http://www.algotest.ru/program/14646.html>

Массив случайных чисел:

9 74 26 95 67 65 78 40 94 57 41 98 83 78 75 26 55 57 74 20 19 82 17 82 9 61 23 36 59 95 12 27 31 27 99 60 94 30 89 2

Сортировка пузырьком:

1. Сортировка неупорядоченного массива из 40 элементов:

Количество сравнений: 780

Количество перестановок: 423

Отсортированный массив:

2 9 9 12 17 19 20 23 26 26 27 27 30 31 36 40 41 55 57 57 59 60 61 65 67 74 74 75 78 78 82 82 83 89 94 94 95 95 98 99

2. Сортировка упорядоченного массива из 40 элементов:

Количество сравнений: 39

Количество перестановок: 0

Отсортированный массив:

2 9 9 12 17 19 20 23 26 26 27 27 30 31 36 40 41 55 57 57 59 60 61 65 67 74 74 75 78 78 82 82 83 89 94 94 95 95 98 99

3. Сортировка упорядоченного массива из 40 элементов с инверсией знаков значений элементов:

Количество сравнений: 780

Количество перестановок: 771

Отсортированный массив:

-99 -98 -95 -95 -94 -94 -89 -83 -82 -82 -78 -78 -75 -74 -74 -67 -65 -61 -60 -59 -57 -57 -55 -41 -40 -36 -31 -30 -27 -27 -26 -26 -23 -20 -19 -17 -12 -9 -9 -2

4. Сортировка неупорядоченного массива из 20 элементов:

Количество сравнений: 190

Количество перестановок: 98

Отсортированный массив:

9 20 26 26 40 41 55 57 57 65 67 74 74 75 78 78 83 94 95 98

5. Сортировка упорядоченного массива из 20 элементов:

Количество сравнений: 19

Количество перестановок: 0

Отсортированный массив:

9 20 26 26 40 41 55 57 57 65 67 74 74 75 78 78 83 94 95 98

6. Сортировка упорядоченного массива из 20 элементов с инверсией знаков значений элементов:

Количество сравнений: 190

Количество перестановок: 186

Отсортированный массив:

-98 -95 -94 -83 -78 -78 -75 -74 -74 -67 -65 -57 -57 -55 -41 -40 -26 -26 -20 -9

7. Сортировка неупорядоченного массива из 10 элементов:

Количество сравнений: 45

Количество перестановок: 19

Отсортированный массив:

9 26 40 57 65 67 74 78 94 95

8. Сортировка упорядоченного массива из 10 элементов:

Количество сравнений: 9

Количество перестановок: 0

Отсортированный массив:

9 26 40 57 65 67 74 78 94 95

9. Сортировка упорядоченного массива из 10 элементов с инверсией знаков значений элементов:

Количество сравнений: 45

Количество перестановок: 45

Отсортированный массив:

-95 -94 -78 -74 -67 -65 -57 -40 -26 -9

Сортировка выбором:

1. Сортировка неупорядоченного массива из 40 элементов:

Количество сравнений: 780

Количество перестановок: 36

Отсортированный массив:

2 9 9 12 17 19 20 23 26 26 27 27 30 31 36 40 41 55 57 57 59 60 61 65 67 74 74 75 78 78 82 82 83 89 94 94 95 95 98 99

2. Сортировка упорядоченного массива из 40 элементов:

Количество сравнений: 780

Количество перестановок: 0

Отсортированный массив:

2 9 9 12 17 19 20 23 26 26 27 27 30 31 36 40 41 55 57 57 59 60 61 65 67 74 74 75 78 78 82 82 83 89 94 94 95 95 98 99

3. Сортировка упорядоченного массива из 40 элементов с инверсией знаков значений элементов:

Количество сравнений: 780

Количество перестановок: 23

Отсортированный массив:

-99 -98 -95 -95 -94 -94 -89 -83 -82 -82 -78 -78 -75 -74 -74 -67 -65 -61 -60 -59 -57 -57 -55 -41 -40 -36 -31 -30 -27 -27 -26 -26 -23 -20 -19 -17 -12 -9 -9 -2

4. Сортировка неупорядоченного массива из 20 элементов:

Количество сравнений: 190

Количество перестановок: 14

Отсортированный массив:

9 20 26 26 40 41 55 57 57 65 67 74 74 75 78 78 83 94 95 98

5. Сортировка упорядоченного массива из 20 элементов:

Количество сравнений: 190

Количество перестановок: 0

Отсортированный массив:

9 20 26 26 40 41 55 57 57 65 67 74 74 75 78 78 83 94 95 98

6. Сортировка упорядоченного массива из 20 элементов с инверсией знаков значений элементов:

Количество сравнений: 190

Количество перестановок: 11

Отсортированный массив:

-98 -95 -94 -83 -78 -78 -75 -74 -74 -67 -65 -57 -57 -55 -41 -40 -26 -26 -20 -9

7. Сортировка неупорядоченного массива из 10 элементов:

Количество сравнений: 45

Количество перестановок: 5

Отсортированный массив:

9 26 40 57 65 67 74 78 94 95

8. Сортировка упорядоченного массива из 10 элементов:

Количество сравнений: 45

Количество перестановок: 0

Отсортированный массив:

9 26 40 57 65 67 74 78 94 95

9. Сортировка упорядоченного массива из 10 элементов с инверсией знаков значений элементов:

Количество сравнений: 45

Количество перестановок: 5

Отсортированный массив:

-95 -94 -78 -74 -67 -65 -57 -40 -26 -9

Для продолжения нажмите любую клавишу . . . ■