Лекция:

Вот теперь программа сортировки точно завершена. Осталось её как следует протестировать.

Лучше всего тестировать программу в обычных и в так называемых «граничных» условиях.

Программе сортировки очень «непривычно» получать на вход *уже* отсортированные по возрастанию или убыванию массивы. Поэтому эти условия для неё граничные. А обычными, даже обыденными, условия будут, если на входе окажутся неотсортированные массивы разной длины.

В программе в качестве переменных циклов теперь используются i и j. Это традиционная «не-разлей-вода» парочка имён переменных, которая часто используется во вложенных циклах.

Задание

1. После цикла выведите в консоль массив usersByDay
2. Измените исходное значение usersByDay на [5, 4, 3, 2, 1, 0]
3. Измените исходное значение usersByDay на [0, 1, 2, 3, 4, 5]
4. Измените исходное значение usersByDay на [812, 1360, 657, 1247]
5. Измените исходное значение usersByDay на [812, 1360, 657, 1247, 165]

Код:

Script.js

let usersByDay = [4, 2, 1, 3];

console.log(usersByDay);

for (let i = 0; i <= usersByDay.length - 2; i++) {

let minValue = usersByDay[i];

for (let j = i + 1; j <= usersByDay.length - 1; j++) {

if (usersByDay[j] < minValue) {

minValue = usersByDay[j];

let swap = usersByDay[i];

usersByDay[i] = minValue;

usersByDay[j] = swap;

}

}

}

Вывод:

[5, 4, 3, 2, 1, 0]

[0, 1, 2, 3, 4, 5]

[0, 1, 2, 3, 4, 5]

[0, 1, 2, 3, 4, 5]

[812, 1360, 657, 1247]

[657, 812, 1247, 1360]

[812, 1360, 657, 1247, 165]

[165, 657, 812, 1247, 1360]

**Решение**

let usersByDay = [812, 1360, 657, 1247, 165];

console.log(usersByDay);

for (let i = 0; i <= usersByDay.length - 2; i++) {

let minValue = usersByDay[i];

for (let j = i + 1; j <= usersByDay.length - 1; j++) {

if (usersByDay[j] < minValue) {

minValue = usersByDay[j];

let swap = usersByDay[i];

usersByDay[i] = minValue;

usersByDay[j] = swap;

}

}

}

console.log(usersByDay);