



Sorting: Bubble Sort ☆

Your Sorting: Bubble Sort submission got 30.00 points.

Share

Tweet



[Try the next challenge](#)

Problem

Submissions

Leaderboard

Editorial

Check out the resources on the page's right side to learn more about bubble sort. The video tutorial is by Gayle Laakmann McDowell, author of the best-selling interview book [Cracking the Coding Interview](#).

Consider the following version of Bubble Sort:

```
for (int i = 0; i < n; i++) {  
    for (int j = 0; j < n - 1; j++) {  
        // Swap adjacent elements if they are in decreasing order  
        if (a[j] > a[j + 1]) {  
            swap(a[j], a[j + 1]);  
        }  
    }  
}
```

Given an array of integers, sort the array in ascending order using the Bubble Sort algorithm above. Once sorted, print the following three lines:

1. Array is sorted in numSwaps swaps., where *numSwaps* is the number of swaps that took place.
2. First Element: firstElement, where *firstElement* is the first element in the sorted array.
3. Last Element: lastElement, where *lastElement* is the last element in the sorted array.

Hint: To complete this challenge, you must add a variable that keeps a running tally of all swaps that occur during execution.

For example, given a worst-case but small array to sort: $a = [6, 4, 1]$ we go through the following steps:

swap	a
0	[6, 4, 1]
1	[4, 6, 1]
2	[4, 1, 6]
3	[1, 4, 6]

It took **3** swaps to sort the array. Output would be

```
Array is sorted in 3 swaps.  
First Element: 1  
Last Element: 6
```



Function Description

Complete the function `countSwaps` in the editor below. It should print the three lines required, then return.

`countSwaps` has the following parameter(s):

- `a`: an array of integers .

Input Format

The first line contains an integer, n , the size of the array a .

The second line contains n space-separated integers $a[i]$.

Constraints

- $2 \leq n \leq 600$
- $1 \leq a[i] \leq 2 \times 10^6$

Output Format

You must print the following three lines of output:

- Array is sorted in `numSwaps` swaps., where *numSwaps* is the number of swaps that took place.
- First Element: `firstElement`, where *firstElement* is the first element in the sorted array.
- Last Element: `lastElement`, where *lastElement* is the last element in the sorted array.

Sample Input 0

```
3
1 2 3
```

Sample Output 0

```
Array is sorted in 0 swaps.
First Element: 1
Last Element: 3
```

Explanation 0

The array is already sorted, so **0** swaps take place and we print the necessary three lines of output shown above.

Sample Input 1

```
3
3 2 1
```

Sample Output 1

```
Array is sorted in 3 swaps.
First Element: 1
Last Element: 3
```

Explanation 1

The array is not sorted, and its initial values are: **{3, 2, 1}**. The following **3** swaps take place:

- {3, 2, 1} → {2, 3, 1}**
- {2, 3, 1} → {2, 1, 3}**
- {2, 1, 3} → {1, 2, 3}**

At this point the array is sorted and we print the necessary three lines of output shown above.

Current Buffer (saved locally, editable)



Java 8



```
1 import java.io.*;
2 import java.math.*;
3 import java.security.*;
4 import java.text.*;
5 import java.util.*;
6 import java.util.concurrent.*;
7 import java.util.regex.*;
8
9 public class Solution {
10
11     // Complete the countSwaps function below.
12     static void countSwaps(int[] a) {
13         int swaps = 0;
14         for(int i = 0; i<a.length; i++) {
15             for(int j = 0; j<a.length-1; j++) {
16                 if(a[j] > a[j+1]) {
17                     int temp = a[j];
18                     a[j] = a[j+1];
19                     a[j+1] = temp;
20                     swaps++;
21                 }
22             }
23         }
24         System.out.println("Array is sorted in " + swaps + " swaps.");
25         System.out.println("First Element: " + a[0]);
26         System.out.println("Last Element: " + a[a.length-1]);
27     }
28
29     private static final Scanner scanner = new Scanner(System.in);
30
31     public static void main(String[] args) {
32         int n = scanner.nextInt();
33         scanner.skip("(\\r\\n|\\n\\r\\u2028\\u2029\\u0085)?");
34
35         int[] a = new int[n];
36
37         String[] aItems = scanner.nextLine().split(" ");
38         scanner.skip("(\\r\\n|\\n\\r\\u2028\\u2029\\u0085)?");
39
40         for (int i = 0; i < n; i++) {
41             int aItem = Integer.parseInt(aItems[i]);
42             a[i] = aItem;
43         }
44
45         countSwaps(a);
46
47         scanner.close();
48     }
49 }
50
```

[⤶ Upload Code as File](#)☐ [Test against custom input](#)[Run Code](#)[Submit Code](#)

Congratulations

You solved this challenge. Would you like to challenge your friends?

[Next Challenge](#)[✔ Testcase 0](#)[✔ Testcase 1](#)[✔ Testcase 2](#)[✔ Testcase 3](#)

Input (stdin)

[Download](#)

```
3
1 2 3
```

Expected Output

[Download](#)

```
Array is sorted in 0 swaps.
First Element: 1
Last Element: 3
```

Compiler Message

Success