

## Review

2 min

Excellent work! You've successfully implemented your first sorting

Preview: Docs Loading link description

[algorithm](#)

.

## Instructions

1. Checkpoint 1 Passed

1.

Test your work by calling `bubbleSort()` with each of the arrays below and logging the results:

[9, 8, 7, 6, 5, 4, 3, 2, 1]

[1, 2, 3, 4, 5, 6, 7, 8, 9]

Notice how your implementation of bubble sort handles swapping elements in these two arrays. Do the number of swaps differ between the two arrays? Why would that be?

Hint

Don't forget to use `console.log()` to see the result of your calls to `bubbleSort()`.

2. Checkpoint 2 Passed

2.

What real-life examples of bubble sort can you come up with?

Hint

Cars overtaking one another on a one-lane highway illustrate the comparison and swap of bubble sort. If one car is faster than the car in front of it, the first car will "swap" positions and overtake the other car. This might repeat until the fastest car overtakes all other cars in front of it.

3. Checkpoint 3 Passed

3.

Can you think of an input array that would result in an  $O(n^2)$  runtime? What input array would result in an  $O(n)$  runtime?

Hint

An array with the elements already sorted from smallest to greatest would only need to be looped over one time since it requires no sorting.

An array with the elements reverse-sorted where the elements sorted from largest to smallest.

### **bubbleSort.js**

```
const swap = require('./swap');

const bubbleSort = input => {
  let swapCount = 0
  let swapping = true;

  while (swapping) {
    swapping = false;
    for (let i = 0; i < input.length - 1; i++) {
      if (input[i] > input[i + 1]) {
        swap(input, i, i + 1);
        swapCount++;
        swapping = true;
      }
    }
  }

  console.log(`Swapped ${swapCount} times`);
  return input;
};

module.exports = bubbleSort;

const result1 = bubbleSort([9, 8, 7, 6, 5, 4, 3, 2, 1]);
console.log(result1);

const result2 = bubbleSort([1, 2, 3, 4, 5, 6, 7, 8, 9]);
```

```
console.log(result2);
```

### **swap.js**

```
const swap = (arr, indexOne, indexTwo) => {  
  const temp = arr[indexTwo];  
  arr[indexTwo] = arr[indexOne];  
  arr[indexOne] = temp;  
};
```

```
module.exports = swap;
```

### **Output-only Terminal**

Output:

Swapped 36 times

[ 1, 2, 3, 4, 5, 6, 7, 8, 9 ]

Swapped 0 times

[ 1, 2, 3, 4, 5, 6, 7, 8, 9 ]