```js
// array to sort
var array = [9, 2, 5, 6, 4, 3, 7, 10, 1, 8];

// swap function helper
function swap(array, i, j) {
  var temp = array[i];
  array[i] = array[j];
  array[j] = temp;
}

// be careful: this is a very basic implementation which is nice to understand the deep principle of bubble sort (going through all comparisons) but it can be greatly improved for performances
function bubbleSortBasic(array) {
  for(var i = 0; i < array.length; i++) {
    for(var j = 1; j < array.length; j++) {
      if(array[j - 1] > array[j]) {
        swap(array, j - 1, j);
      }
    }
  }
  return array;
}

console.log(bubbleSortBasic(array.slice())); // => [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ]

// correct implementation: this is the usual implementation of the bubble sort algorithm. Some loops execution are avoided if not they are not needed
function bubbleSort(array) {
  var swapped;
  do {
    swapped = false;
    for(var i = 0; i < array.length; i++) {
      if(array[i] && array[i + 1] && array[i] > array[i + 1]) {
        swap(array, i, i + 1);
        swapped = true;
      }
    }
  } while(swapped);
  return array;
}

console.log(bubbleSort(array.slice())); // => [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ]

```