

Bubble Sort

Bubble sort is [a sorting algorithm](#) that **compares two adjacent elements and swaps them until they are in the intended order.**

Working of Bubble Sort

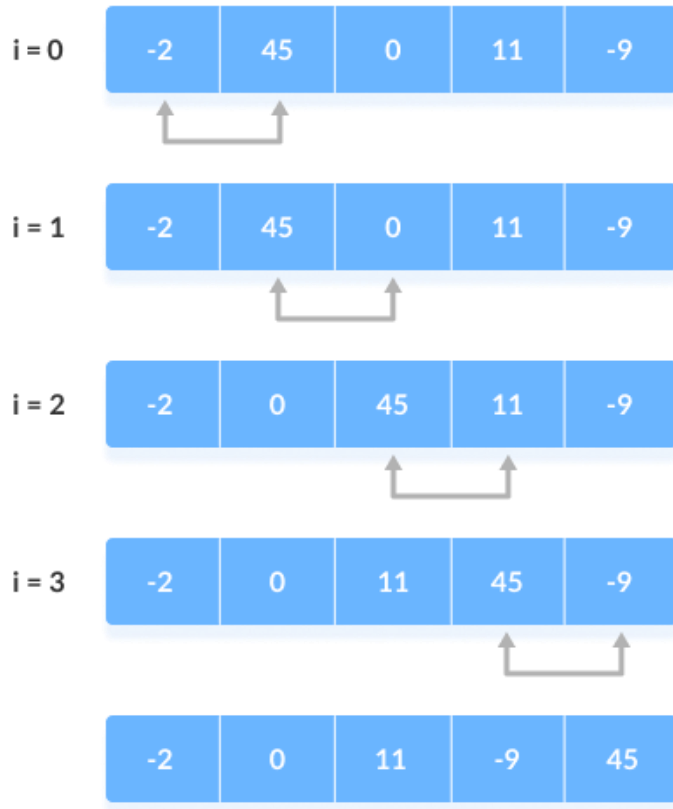
Suppose we are trying to sort the elements in **ascending order.**

1. First Iteration (Compare and Swap)

- 1. Starting from the first index, compare the first and the second elements.**
2. If the first element is greater than the second element, they are swapped.
3. Now, compare the second and the third elements. Swap them if they are not in order.

4. The above process goes on until the last element.

step = 0



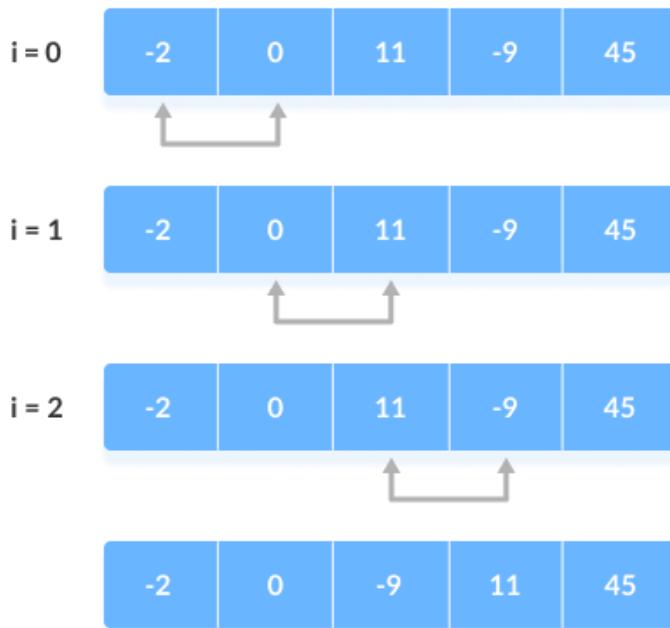
Compare the Adjacent Elements

2. Remaining Iteration

The same process goes on for the remaining iterations.

After each iteration, the largest element among the unsorted elements is placed at the end.

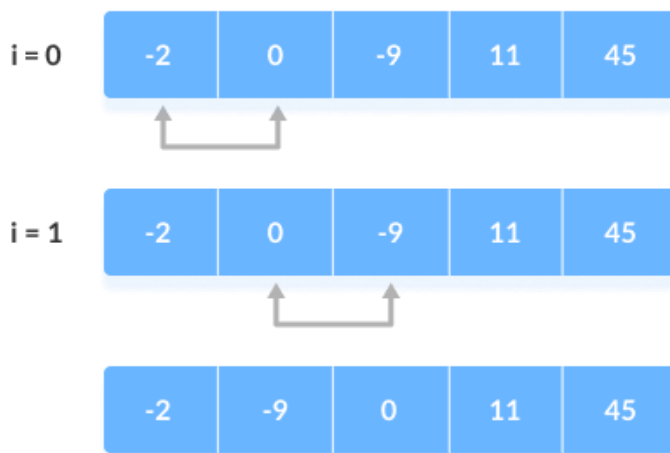
step = 1



Put the largest element at the end

In each iteration, the comparison takes place up to the last unsorted element.

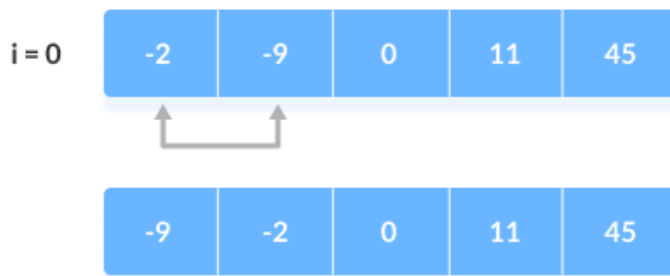
step = 2



Compare the adjacent elements

The array is sorted when all the unsorted elements are placed at their correct positions.

step = 3



The array is sorted if all elements are kept in the right order

Bubble Sort Algorithm

```
bubbleSort(array)
  for i <- 1 to indexOfLastUnsortedElement-1
    if leftElement > rightElement
      swap leftElement and rightElement
  end bubbleSort
```

Optimized Bubble Sort Algorithm

In the above algorithm, all the comparisons are made even if the array is already sorted.

This increases the execution time.