Non-Decreasing Order

If each element in a sequence of values is bigger than or equal to the one before it, the sequence is said to be in non-decreasing order. When the sequence has duplicate values, this order happens. For instance, the elements 1, 3, 3, 6, and 8 are not in decreasing order since each one is bigger than or equal to (in the case of element 3) but not lower than the one before it.

Data Structure - Bubble Sort Algorithm A straightforward sorting algorithm is bubble sort Each pair of adjacent elements in this comparison based sorting algorithm is compared to each other, and it mey are not in the correct order, the elements are wapped. This algorithm's average and worstcase complexity, where one the number of items, is O(n2), making it unsuitable for huge data sets.

How Bubble Sort Works?

We take an unsorted array for our example. Bubble sort takes $O(n^2)$ time so we're keeping it short and precise.



Bubble sort starts with very first two elements, comparing them to check which one is greater.

14	33	27	35	10

It swaps 33 with 27. It also checks with all the elements of sorted sub-list. Here we see that the sorted sub-list has only one element 14, and 27 is greater than 14. Hence, the sorted sub-list remains sorted after swapping.



By now we have 14 and 27 in the sorted sub-list. Next, it compares 33 with 10.





However, swapping makes 27 and 10 unsorted.



Hence, we swap them too.



Again we find 14 and 10 in an unsorted order.





After two iterations, two least values are positioned at the beginning in a sorted manner.



The same process is applied to the rest of the items in the array.

Following is a pictorial depiction of the entire sorting process –

