Sorting

Sorting refers to arranging data in a particular format. Sorting algorithm specifies the way to arrange data in a particular order. Most common orders are in numerical or lexicographical order.

The importance of sorting lies in the fact that data searching can be optimized to a very high level , if data is stored in a sorted manner.

Sorting is also used to represent data in more readable formats.

Characterstics of sorting algorithms

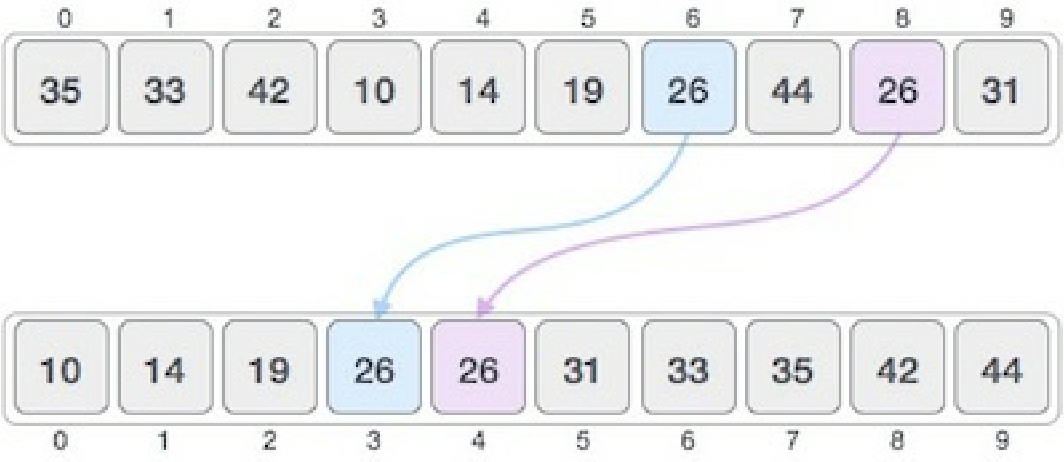
1. In-place Sorting and Not-in-place Sorting

Sorting algorithms may require some extra space for comparison and temporary storage of few data elements. These algorithms do not require any extra space and **sorting is said to happen in-place, or for example, within the array itself**. This is called **in-place sorting**. Bubble sort is an example of in-place sorting.

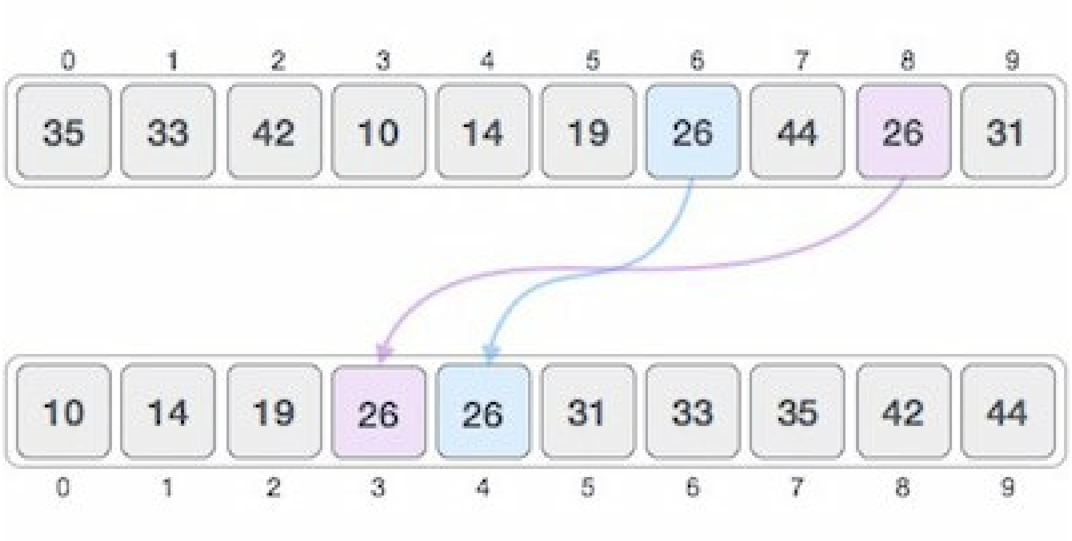
However, in some sorting algorithms, the program requires space which is more than or equal to the elements being sorted. **Sorting which uses equal or more space is called not-in-place sorting**. Merge-sort is an example of not-in-place sorting.

2. Stable and Not Stable Sorting

If a sorting algorithm, after sorting the contents, **does not change the sequence of similar content** in which they appear, it is called **stable sorting.**



If a sorting algorithm, after sorting the contents, **changes the sequence of similar content in which they appear**, it is called **unstable sorting**.



3. Adaptive and Non-Adaptive Sorting Algorithm

A sorting algorithm is said to be **adaptive, if it takes advantage of**

already ‘sorted’ elements in the list that is to be sorted. That is, while sorting if the source list has some element already sorted, adaptive algorithms will take this into account and will try not to re-order them.

A **non-adaptive algorithm** is one which **does not take into account the elements which are already sorted**. They try to force every single element to be re-ordered to confirm their sortedness.

Types of Sorting

There are a lot of sorting algorithms & their variations present in usage. We will be characterizing them in to 2 categories for now

1. Elementary Sorting(Mostly Average time complexity>=O(n2))
   - Bubble sort
   - Selection sort
   - Insertion sort
2. Advanced Sorting
   - Quick sort
   - Merge sort
   - Heap sort
   - Shell sort ...

Please go through these articles to understand how these sorts work and then head to assignments

Bubble sort - Link

Selection sort - Link

Insertion sort - Link

`

Further Reading(Optional)

MIT Sorting Lecture Content

(Make sure to read the last section which explains **“How to select a sorting algorithm”**)

Geekforgeek - Time complexity of all the sorting algoritms