Sorting Algorithms

Sorting Algorithm

- A **Sorting Algorithm** is used to rearrange a given array or list of elements in an order

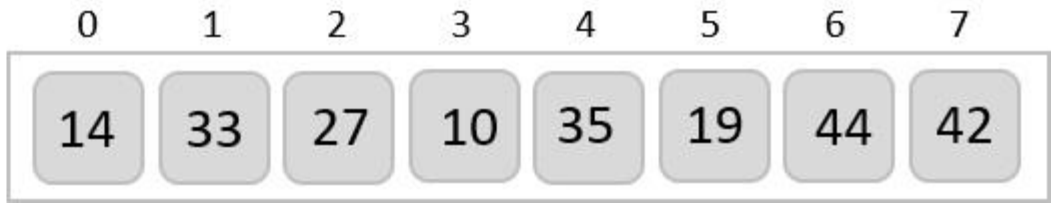
Bubble Sort

- Bubble sort is a simple sorting algorithm.
- This sorting algorithm is comparison-based algorithm.
- each pair of adjacent elements is compared and the elements are swapped if they are not in order.

Bubble Sort (Steps)

- **Step 1** − Check if the first element in the input array is greater than the next element in the array.
- **Step 2** − If it is greater, swap the two elements; otherwise move the pointer forward in the array.
- **Step 3** − Repeat Step 2 until we reach the end of the array.
- **Step 4** − Check if the elements are sorted; if not, repeat the same process (Step 1 to Step 3) from the last element of the array to the first.
- **Step 5** − The final output achieved is the sorted array.

Bubble Sort (Example)



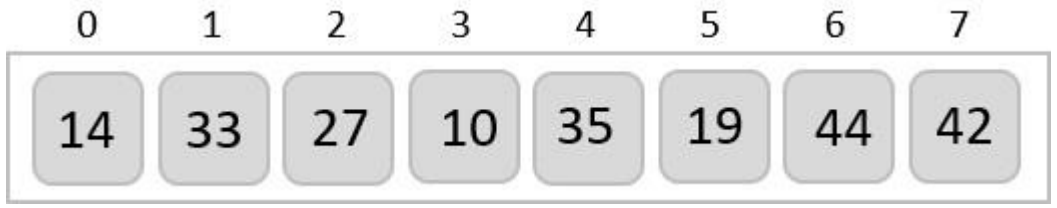
Insertion Sort

- This is an in-place comparison-based sorting algorithm.
- a sub-list is maintained which is always sorted.
- An element which is to be 'inserted' in this sorted sub-list, has to find its appropriate place and then it has to be inserted there.

Insertion Sort(Steps)

- **Step 1** – If it is the first element, it is already sorted. return 1;
- **Step 2** – Pick next element
- **Step 3** – Compare with all elements in the sorted sub-list
- **Step 4** – Shift all the elements in the sorted sub-list that is greater than the value to be sorted
- **Step 5** – Insert the value
- **Step 6** – Repeat until list is sorted

Insertion Sort (Example)



Selection Sort

- An in-place comparison-based algorithm.
- The smallest element is selected from the unsorted array and swapped with the leftmost element.

Selection Sort (Steps)

Step 1. Set MIN to location 0.

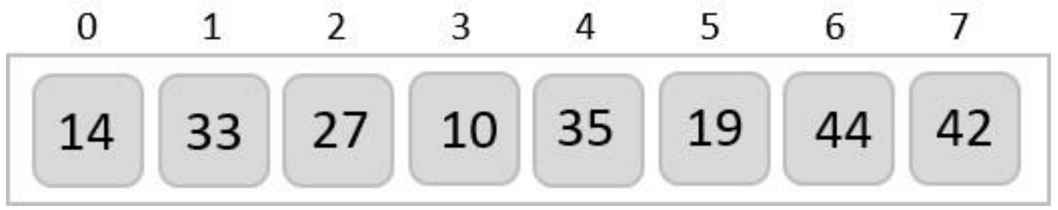
Step 2. Search the minimum element in the list.

Step 3. Swap with value at location MIN.

Step 4. Increment MIN to point to next element.

Step 5. Repeat until the list is sorted.

Selection Sort (Example)


0
1
2
3
4
5
6
7
14
33
27
10
35
19
44
42