

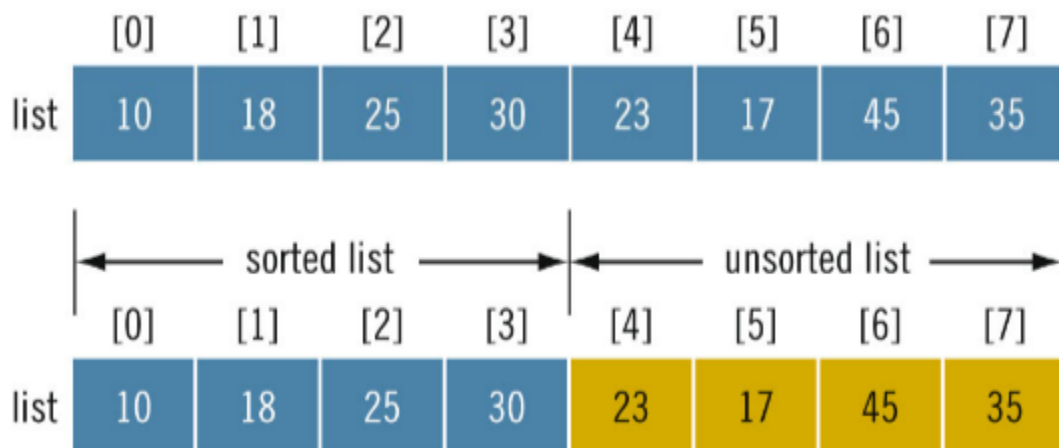
Insertion Sort Info:

1) How does it work?:

- Similar to the bubble sort algorithm we: Compare adjacent elements, swap the elements if **array[index] < array[index-1]**, the array is divided into two sublists of sorted and unsorted arrays, and so fourth.
- Unlike the bubble sort algorithm, insertion sort will compare the element behind the current element and will continually compare the adjacent element behind. Of course, it will continue to do so until the condition in the while loop becomes false.
- Let us recall in bubble sort we continuously compared the adjacent element forward (on the right). However, we would reduce the number of elements/indices on the right to eventually it being the first two indices. Whereas in insertion sort, not only are we gradually increasing in the current index we are in, but we are also continuously increasing the number of adjacent elements we compare on the left (behind the current element).

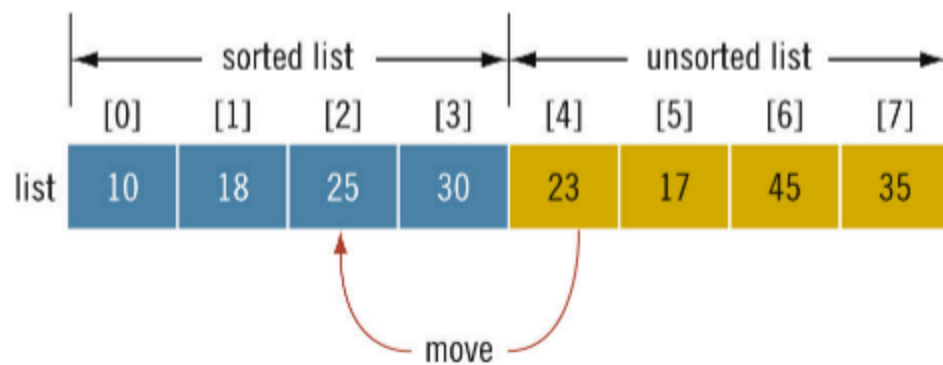
2) Illustrations:

- Sorts the list by moving each element to its proper place

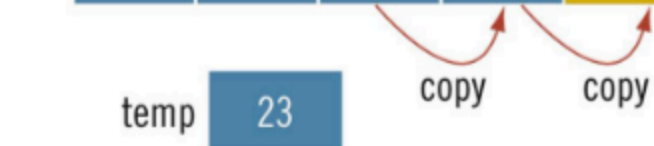
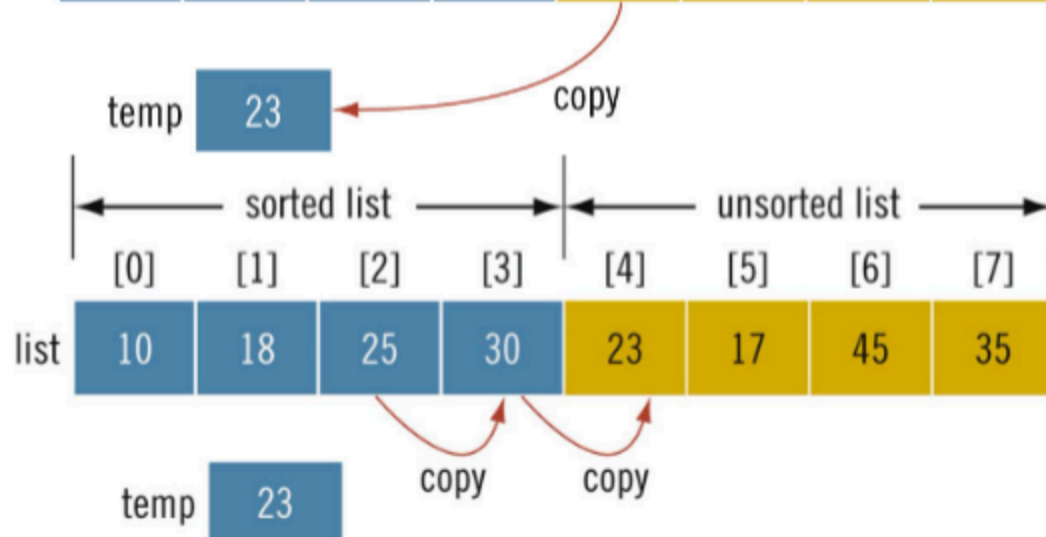


Insertion Sort (2)

- Consider the element `list[4]`
 - First element of unsorted list
 - Move `list[4]` to proper location at `list[2]`



Insertion Sort (3)



Insertion Sort (4)

