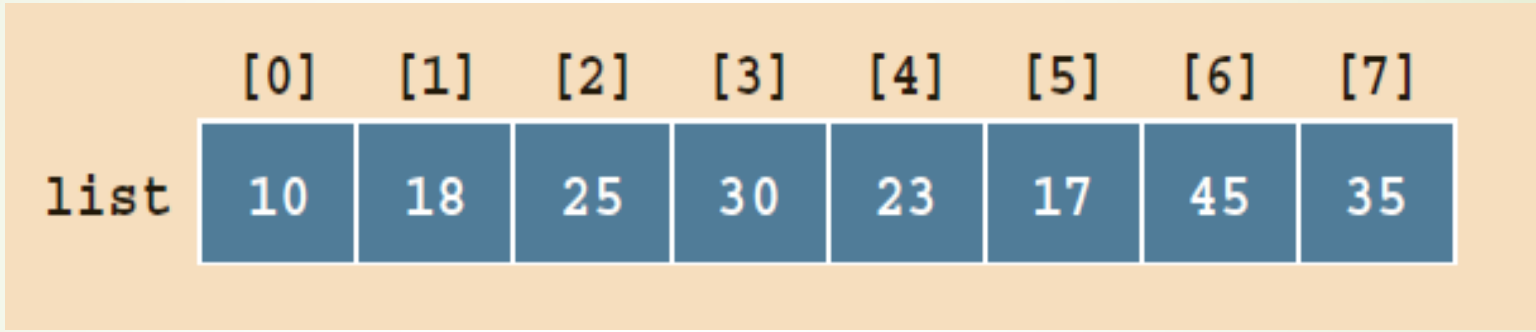
Programming Fundamentals

Aamina Batool

Insertion Sort

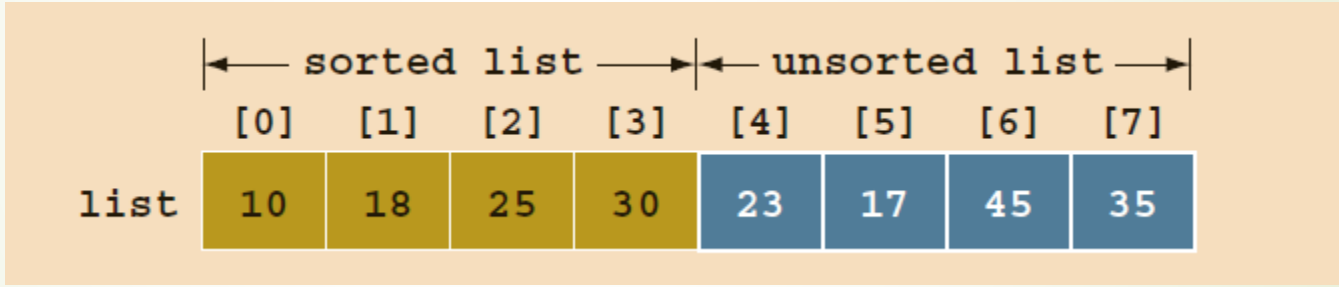
- The insertion sort algorithm sorts the list by moving each element to its proper place in the **sorted portion of the list**.

Example



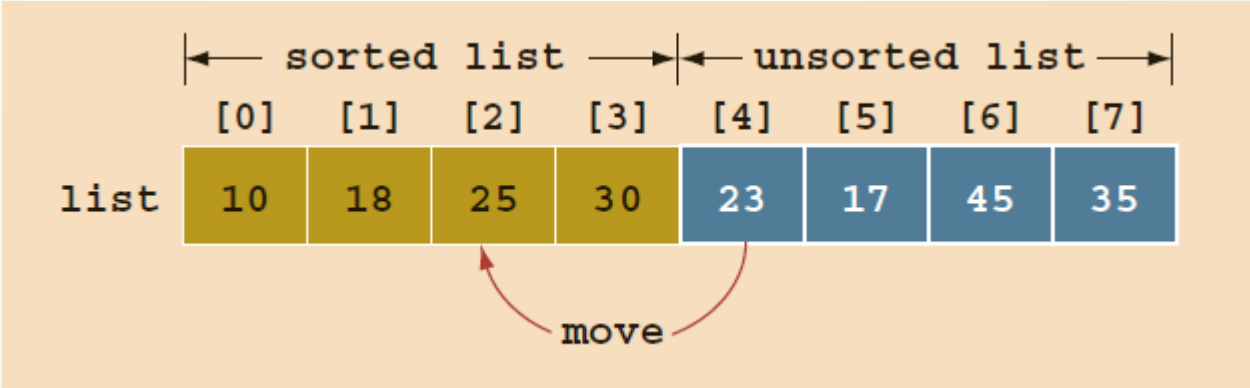
Example

- The length of the list is **8**. In this list, the elements **`list[0]`**, **`list[1]`**, **`list[2]`**, and **`list[3]`** are in order. That is, **`list[0]...list[3]`** is sorted



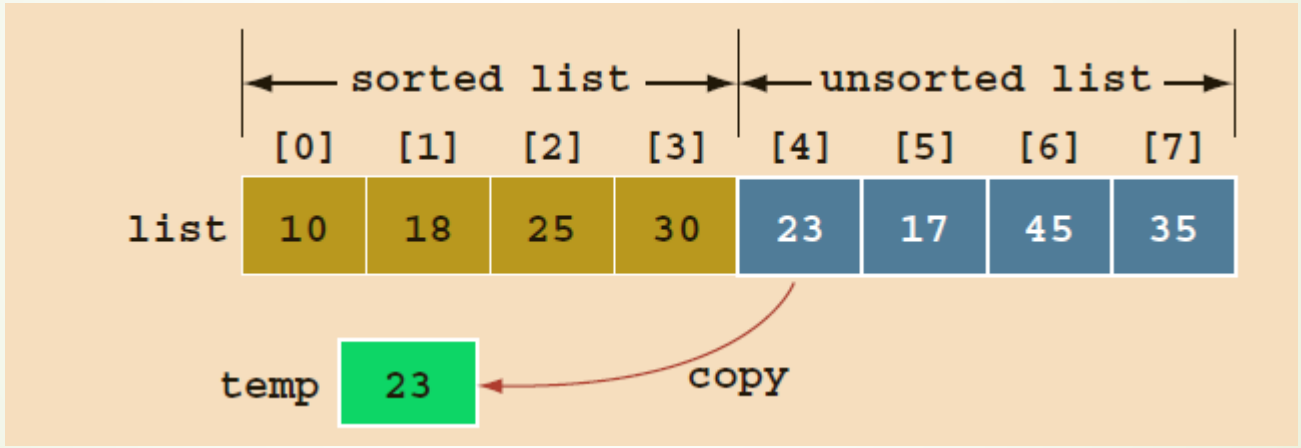
Example

- Next, we consider the element **`list[4]`**, the first element of the unsorted list. Because **`list[4] < list[3]`**, we need to move the element **`list[4]`** to its proper location. It thus follows that element **`list[4]`** should be moved to **`list[2]`**



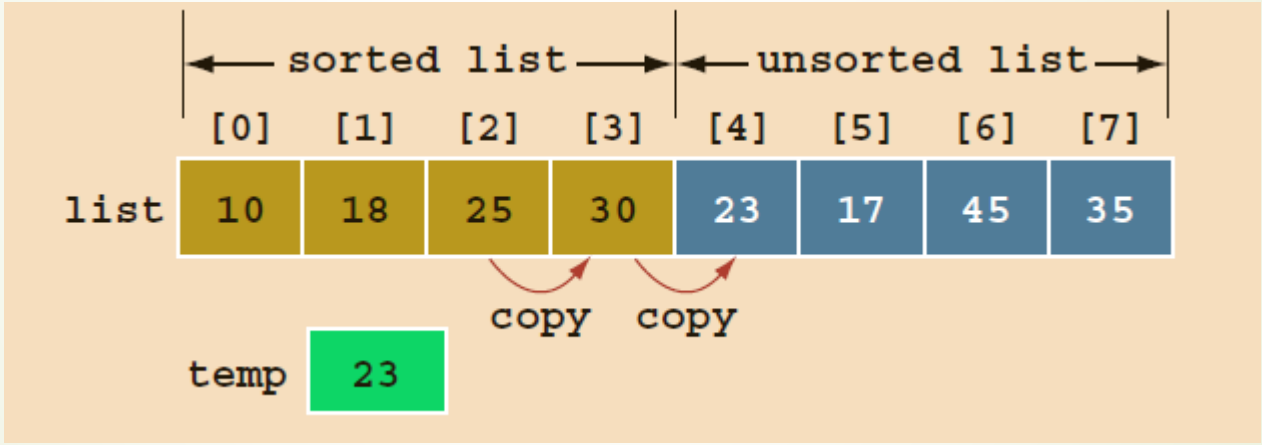
Example

- To move **`list[4]`** into **`list[2]`**, first we copy **`list[4]`** into **`temp`**, a temporary memory space



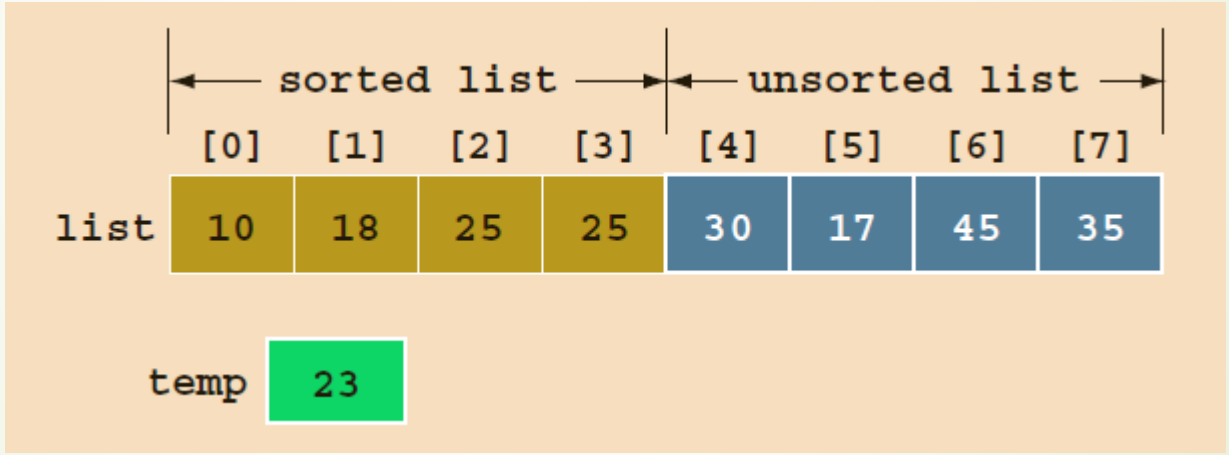
Example

- Next, we copy **`list[3]`** into **`list[4]`** and then **`list[2]`** into **`list[3]`**



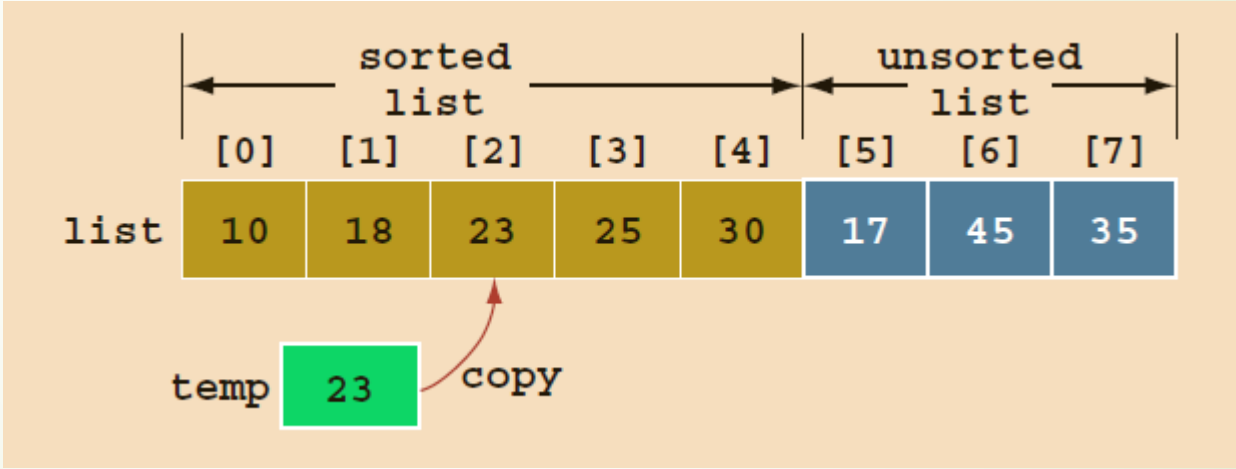
Example

- After copying **`list[3]`** into **`list[4]`** and **`list[2]`** into **`list[3]`**, the list is as shown in Figure



Example

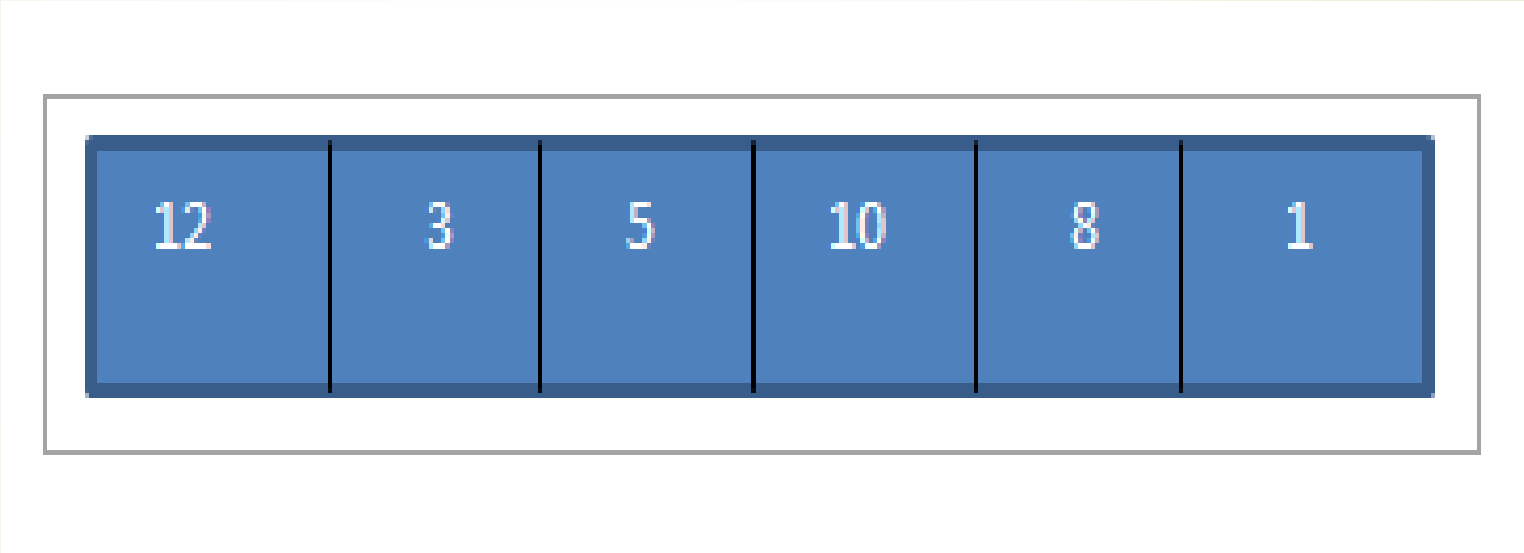
- We now copy **`temp`** into **`list[2]`**.



Insertion Sort

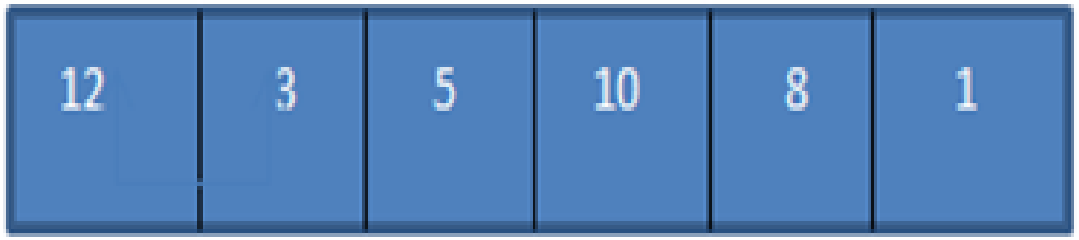
- Now **`list[0]...list[4]`** is sorted, and **`list[5]...list[7]`** is unsorted
- we see that during the sorting phase, the array containing the list is divided into two sublists: *sorted* and *unsorted*.
- Elements in the sorted sublist are in order
- Elements in the unsorted sublist are to be moved one at a time to their proper places in the sorted sublist.

Example



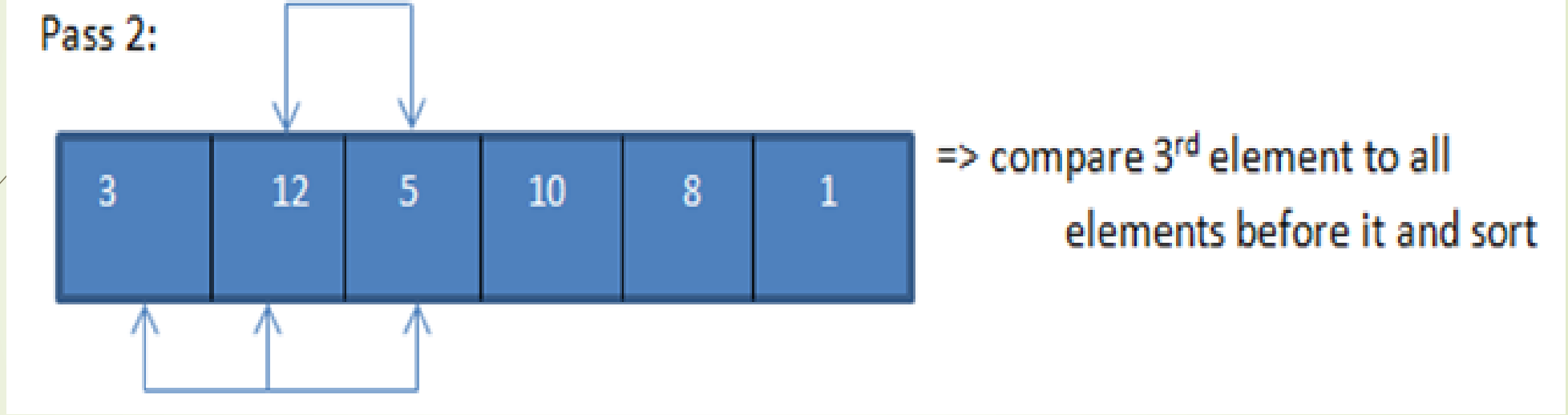
Example

Pass 1:



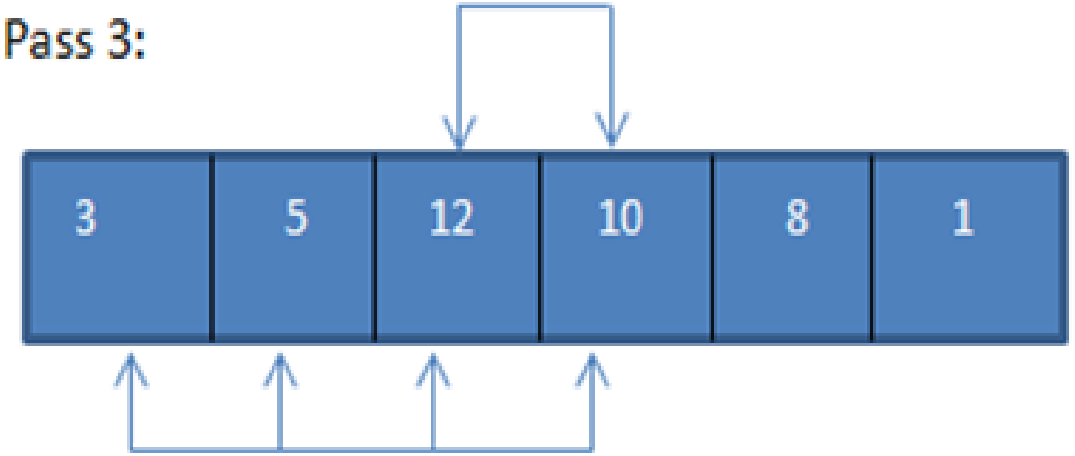
=> compare 2nd element to 1st element before it and sort

Example


Pass 2:
3
12
5
10
8
1
=> compare 3rd element to all
elements before it and sort

Example

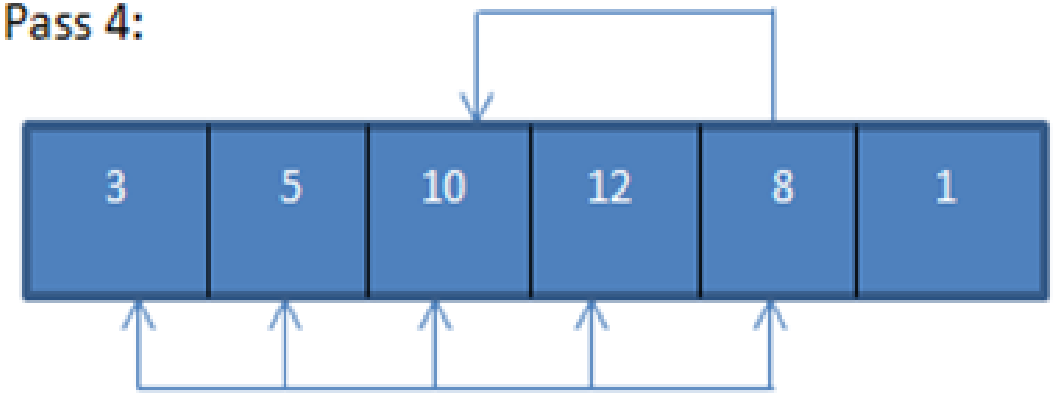
Pass 3:



=> compare 4th element to all elements before it and sort

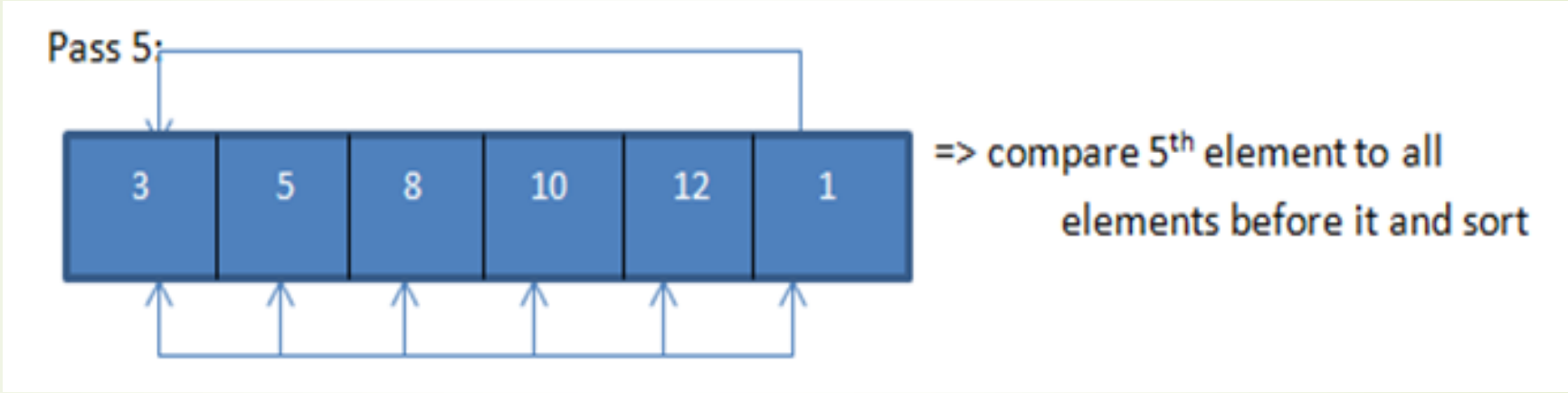
Example

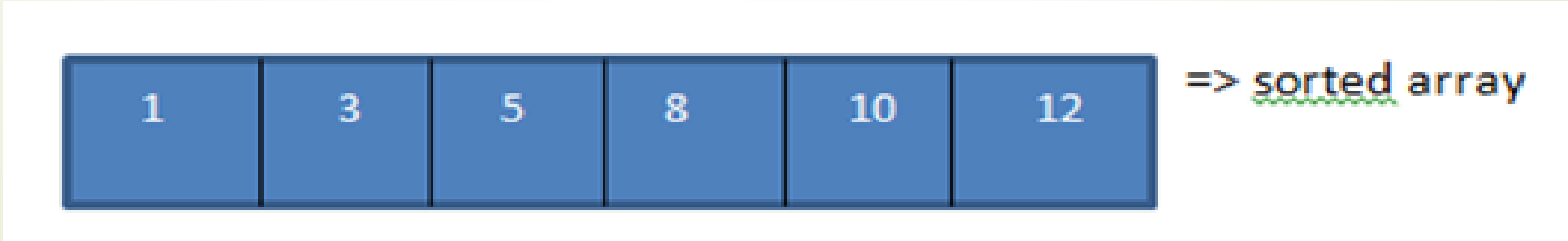
Pass 4:



=> compare 5th element to all elements before it and sort

Example


Pass 5:
3
5
8
10
12
1
=> compare 5th element to all elements before it and sort


1
3
5
8
10
12
=> sorted array

Exercise

- Implement Insertion Sort for sorting an array in:
- Ascending order
- Descending order

Insertion Sort - Algorithm

```
void insertionSort(elemType list[], int length)
{
    for (int firstOutOfOrder = 1; firstOutOfOrder < length;
                                  firstOutOfOrder++)
        if (list[firstOutOfOrder] < list[firstOutOfOrder - 1])
        {
            elemType temp = list[firstOutOfOrder];
            int location = firstOutOfOrder;

            do
            {
                list[location] = list[location - 1];
                location--;
            }
            while(location > 0 && list[location - 1] > temp);

            list[location] = temp;
        }
} //end insertionSort
```

References

1. C++ Programming: From Problem Analysis to Program Design, Third Edition
2. https://www.just.edu.jo/~yahya-t/cs115/
3. https://www.softwaretestinghelp.com/insertion-sort/