**SORTING**

Sorting refers to arranging data in a particular order (ascending or descending order).

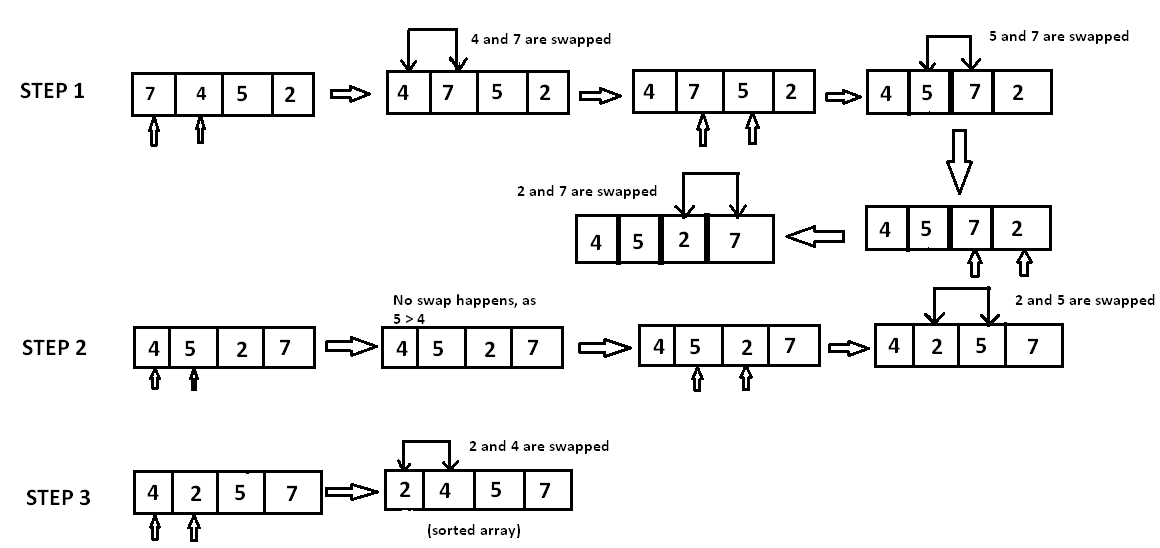
Types of Sorting :

1. Bubble sort
2. Selection sort
3. Insertion sort
4. Quick sort
5. Merge sort

**1.Bubble sort**

Bubble sort is based on the idea of repeatedly comparing pairs of adjacent elements and then swapping their positions if they exist in the wrong order.

Lets try to understand with an example: A [ ] = { 7, 4, 5, 2}



In step 1, 7 is compared with 4. Since 7>4, 7 is moved ahead of 4. Since all the other elements are of a lesser value than 7, 7 is moved to the end of the array.

Now the array is A[]={4,5,2,7}.

In step 2, 4 is compared with 5. Since 5>4 and both 4 and 5 are in ascending order, these elements are not swapped. However, when 5 is compared with 2, 5>2 and these elements are in descending order. Therefore, 5and 2 are swapped.

Now the array is A[]={4,2,5,7}.

In step 3, the element 4 is compared with 2. Since 4>2 and the elements are in descending order, 4 and 2 are swapped.

The sorted array is A[]={2,4,5,7}.

**2.Selection sort**

Selection Sort algorithm is used to arrange a list of elements in a particular order (Ascending or Descending). In selection sort, the first element in the list is selected and it is compared repeatedly with remaining all the elements in the list. If any element is smaller than the selected element (for Ascending order), then both are swapped. Then we select the element at second position in the list and it is compared with remaining all elements in the list. If any element is smaller than the selected element, then both are swapped. This procedure is repeated till the entire list is sorted.

**The selection sort algorithm is performed using following steps...**

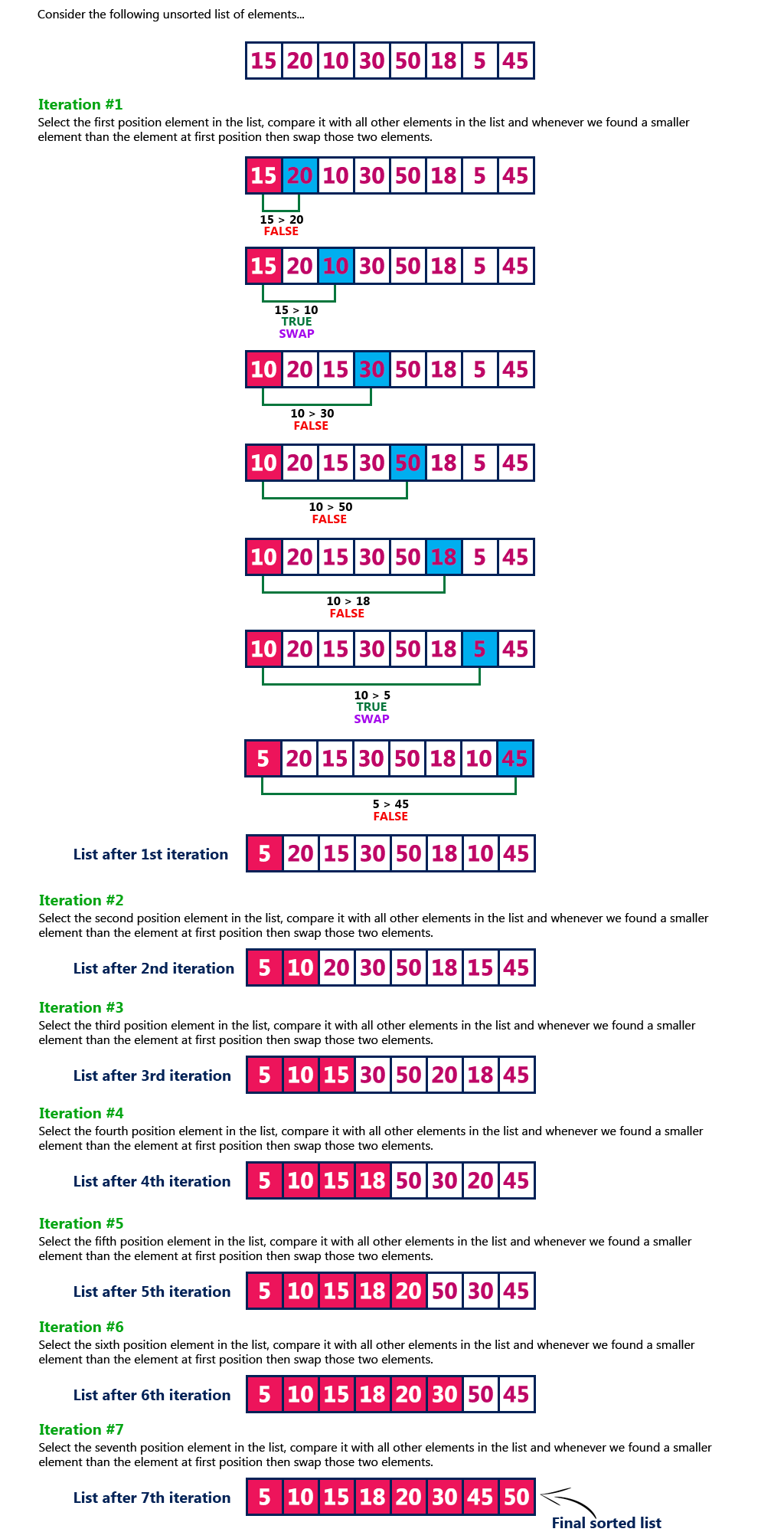
Step 1: Select the first element of the list (i.e., Element at first position in the list).

Step 2: Compare the selected element with all other elements in the list.

Step 3: For every comparison, if any element is smaller than selected element (for Ascending order), then these two are swapped.

Step 4: Repeat the same procedure with next position in the list till the entire list is sorted.

**Example of selection sort**

****

**3.Insertion Sort**

Insertion sort algorithm arranges a list of elements in a particular order. In insertion sort algorithm, every iteration moves an element from unsorted portion to sorted portion until all the elements are sorted in the list.

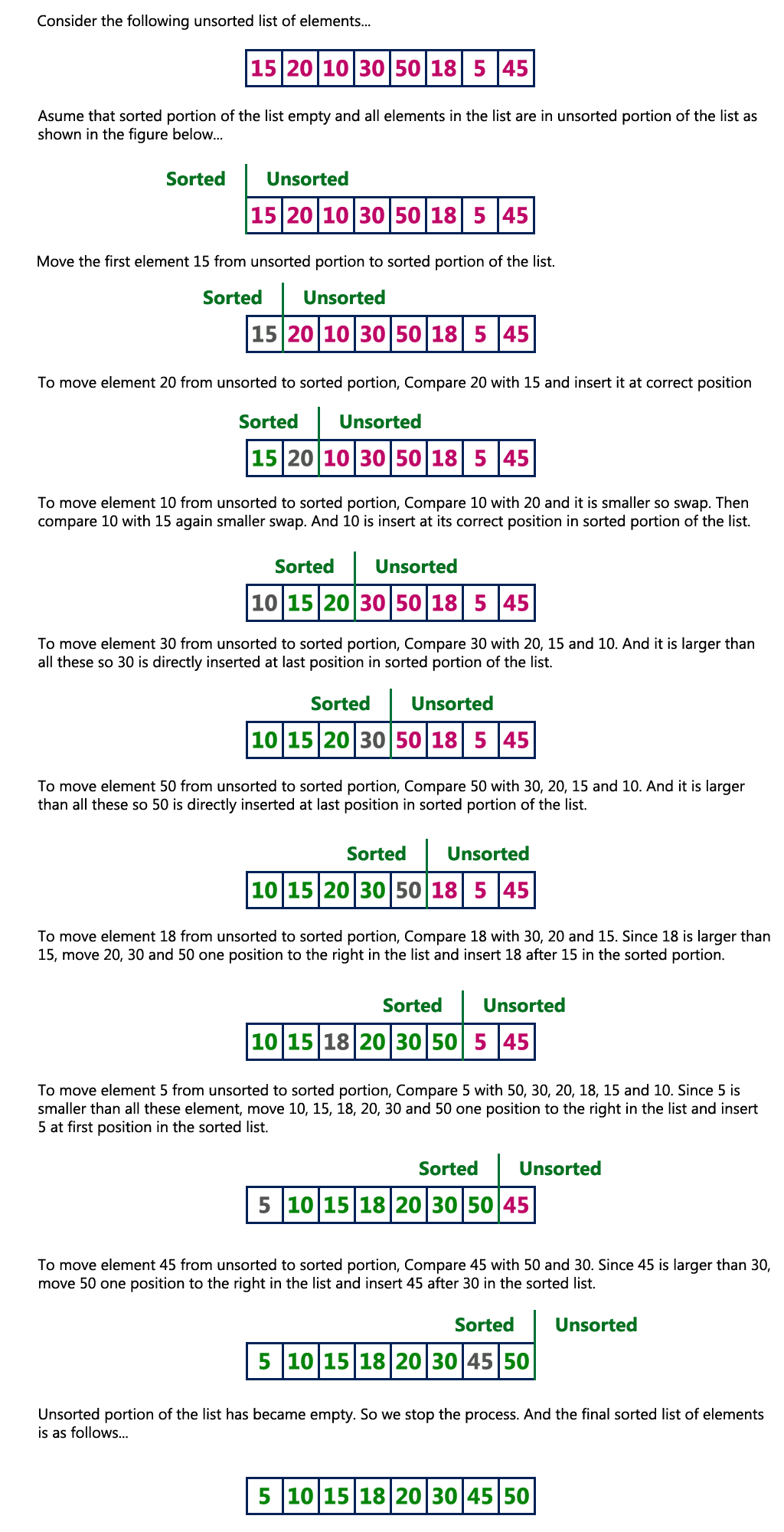
**The insertion sort algorithm is performed using following steps...**

Step 1: Asume that first element in the list is in sorted portion of the list and remaining all elements are in unsorted portion.

Step 2: Consider first element from the unsorted list and insert that element into the sorted list in order specified.

Step 3: Repeat the above process until all the elements from the unsorted list are moved into the sorted list.

**Example of Insertiom sort**

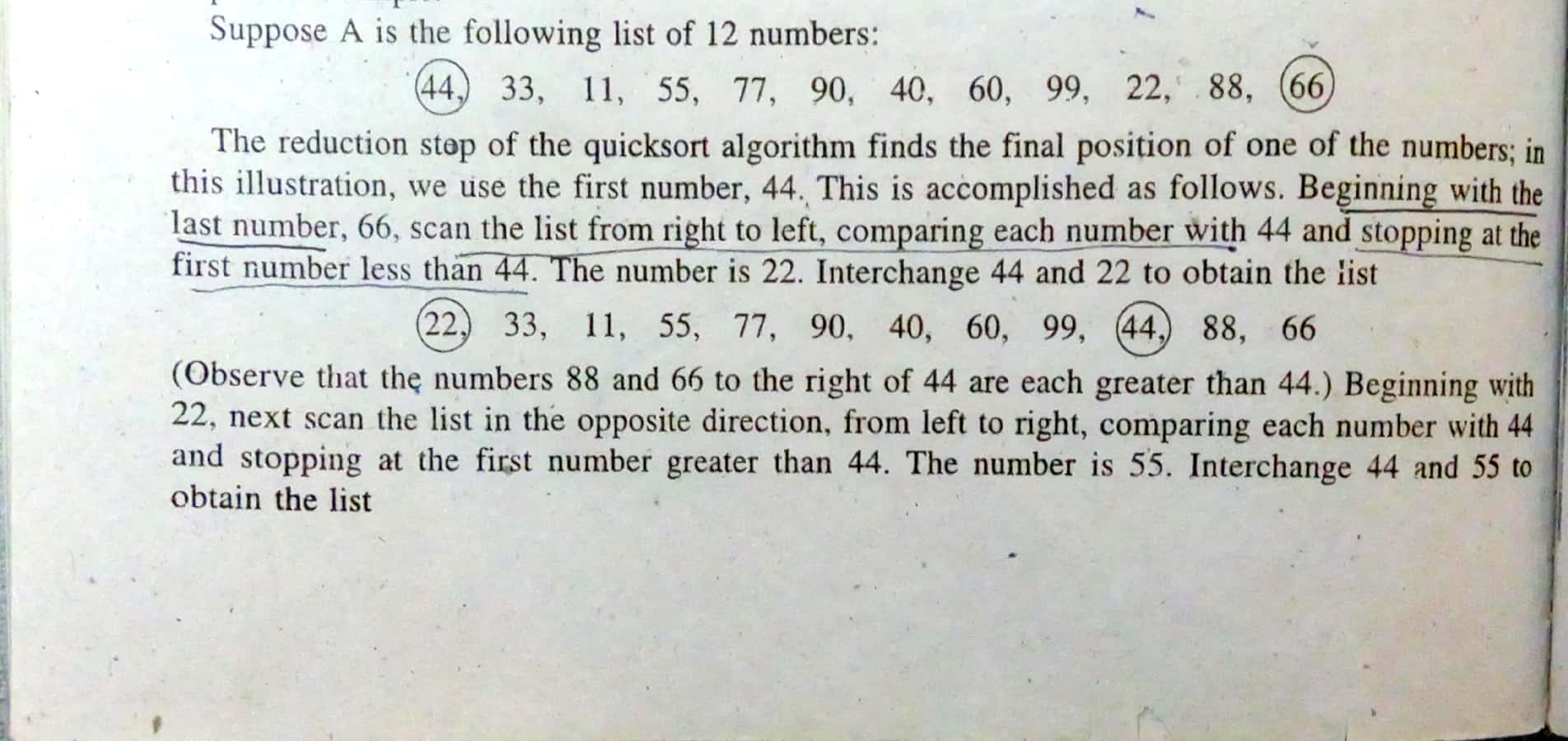


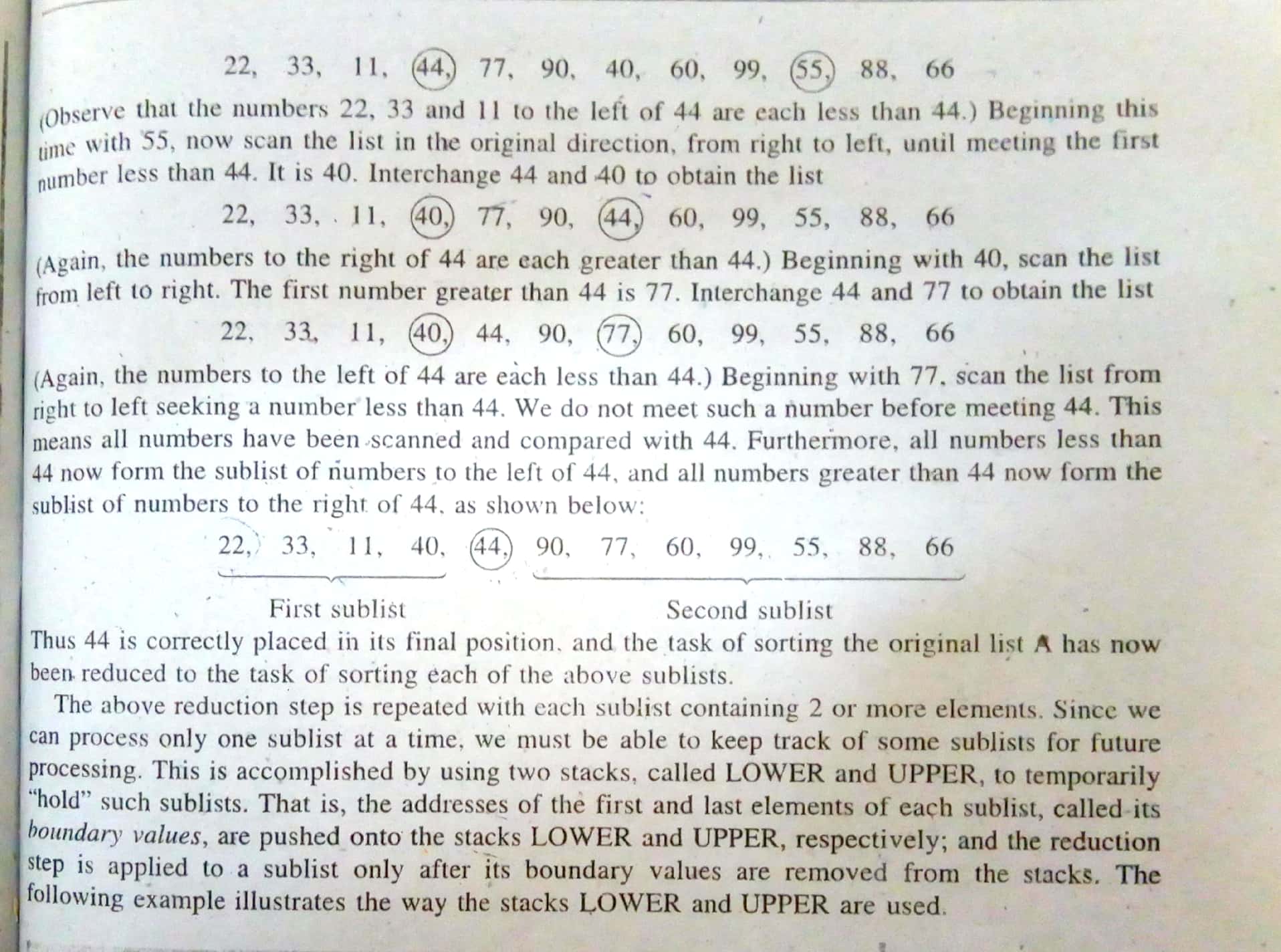
**4.Quick sort**

Quick sort is a highly efficient sorting algorithm and is based on partitioning of array of data into smaller arrays. A large array is partitioned into two arrays one of which holds values smaller than the specified value, say pivot, based on which the partition is made and another array holds values greater than the pivot value. QuickSort is a Divide and Conquer algorithm. It picks an element as pivot and partitions the given array around the picked pivot. There are many different versions of quickSort that pick pivot in different ways.

1. Always pick first element as pivot.
2. Always pick last element as pivot (implemented below)
3. Pick a random element as pivot.
4. Pick median as pivot.

**Example of quick sort**

****



**NOTE – For all the sorting programs refer the txt files.**