Merge Sort

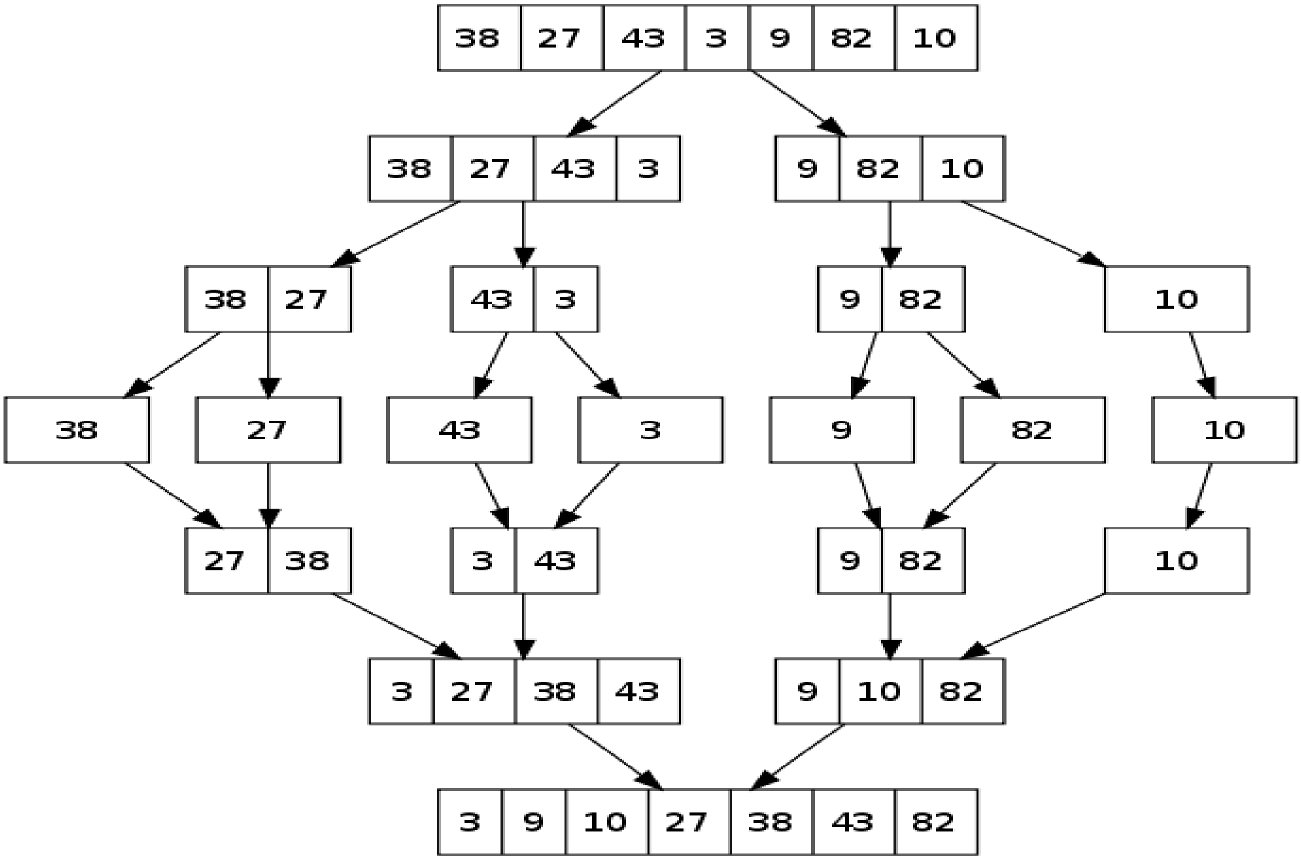
Merge Sort follows the rule of Divide and Conquer. In merge sort the unsorted list is divided into N sublists, each having one element, because a list consisting of one element is always sorted. Then, it repeatedly merges these sublists, to produce new sorted sublists, and in the end, only one sorted list is produced.

Example

Let us take an example of the following elements

38 27 43 3 9 82 10

Here apply the Merge sort algorithm to sort the elements.



Example	Output

def merge_sort(list):
    if len(list)>1:
        mid = len(list)//2
        left = list[:mid]
        right = list[mid:]
 	
        merge_sort(left)
        merge_sort(right)
 	
        i=0
        j=0
        k=0
        while i < len(left) and j < len(right):
            if left[i] < right[j]:
                list[k]=left[i]
                i=i+1
            else:
                list[k]=right[j]
                j=j+1
            k=k+1
 	
        while i < len(left):
	3, 9, 10, 27, 38, 43, 82

```
            list[k]=left[i]
            i=i+1
            k=k+1

        while j < len(right):
            list[k]=right[j]
            j=j+1
            k=k+1

list = [38,27,43,3,9,82,10]
merge_sort(list)
print(list)
```