

80. merge sort

Program:

```
def merge_sort(arr):
    if len(arr) > 1:
        mid = len(arr) // 2
        L = arr[:mid]
        R = arr[mid:]

        merge_sort(L)
        merge_sort(R)

        i = j = k = 0

        while i < len(L) and j < len(R):
            if L[i] < R[j]:
                arr[k] = L[i]
                i += 1
            else:
                arr[k] = R[j]
                j += 1
            k += 1

        while i < len(L):
            arr[k] = L[i]
            i += 1
            k += 1

        while j < len(R):
            arr[k] = R[j]
            j += 1
            k += 1

    return arr
```

Example

```
arr = [12, 11, 13, 5, 6, 7]
print("Given array is", arr)
result = merge_sort(arr)
print("Sorted array is", result)
```

Output:

```
Given array is [12, 11, 13, 5, 6, 7]
Sorted array is [5, 6, 7, 11, 12, 13]
```

```
=== Code Execution Successful ===
```

Time complexity: $O(n \log n)$