Merge Sort

Merge sort is a sorting technique based on divide and conquer technique. With worst-case time complexity being **O(n log n)**, it is one of the most used and approached algorithms. Merge sort first divides the array into equal halves and then combines them in a sorted manner.

What Is a Divide and Conquer Algorithm?

Divide-and-conquer recursively solves subproblems; each subproblem must be smaller than the original problem, and each must have a base case. A divide-and-conquer algorithm has three parts:

Divide: Divide the list or array recursively into two halves until it can no more be divided.

Conquer: Each subarray is sorted individually using the merge sort algorithm.

Merge: The sorted subarrays are merged back together in sorted order. The process continues until all elements from both subarrays have been merged.

Merge Sort Algorithm

```
MERGE-SORT(A, p, r)
1  if p < r
2      q = ⌊(p + r)/2⌋
3      MERGE-SORT(A, p, q)
4      MERGE-SORT(A, q + 1, r)
5      MERGE(A, p, q, r)
```

```
MERGE(A, p, q, r)
 1  n1 = q − p + 1
 2  n2 = r − q
 3  let L[1 . . n1 + 1] and R[1 . . n2 + 1] be new arrays
 4  for i = 1 to n1
 5      L[i] = A[p + i − 1]
 6  for j = 1 to n2
 7      R[j] = A[q + j]
 8  L[n1 + 1] = ∞
 9  R[n2 + 1] = ∞
10  i = 1
11  j = 1
12  for k = p to r
13      if L[i] ≤ R[j]
14          A[k] = L[i]
15          i = i + 1
16      else A[k] = R[j]
17          j = j + 1
```

Example: A=[2, 4, 5, 7, 1, 2, 3, 6]