

# What is Merge Sort?

Merge sort is one of the most important sorting algorithms based on the divide and conquer technique. Merge sort is very popular for its efficiency to sort the data in a small amount of time. It is one of the best examples of applications for divide and conquers algorithms. Basically, the principle working of merge sort is that it divides the list of data into two halves and again calls these subparts to further divide it into two halves. It keeps repeating the process until each subpart of the list contains only one element. Later, it will join these sub-parts of one element each into two elements by sorting them. Again, the subpart containing two elements will be joined with the other two elements after sorting. This process keeps repeating by calling the function recursively until we get the final sorted list of elements.

## Algorithm for Merge Sort

The divide and conquer approach to sort  $n$  numbers using merge sort consists of separating the array  $T$  into two parts where the sizes are almost the same. These two parts are sorted by recursive calls and then merged with the solution of each part while preserving the order. The algorithm considers two temporary arrays  $U$  and  $V$ , into which the original array  $T$  is divided. When the number of elements to be sorted is small, a relatively simple algorithm is used.

'mergeSort' separates the instance into two halves sized sub instances, solves them recursively, and then combines the two sorted half arrays to obtain the solution to the original instance.

## Example

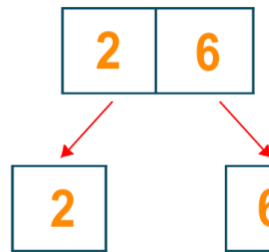
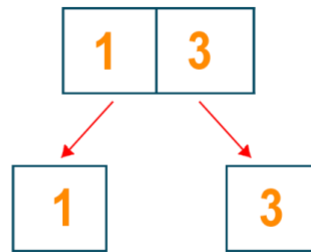
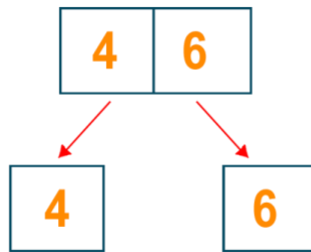
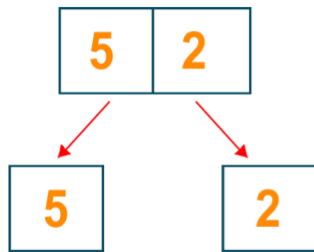
Here we take the given unsorted list of elements.

5	2	4	6	1	3	2	6
---	---	---	---	---	---	---	---

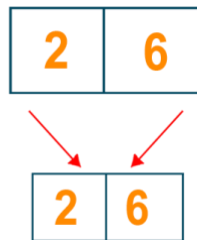
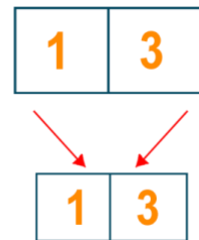
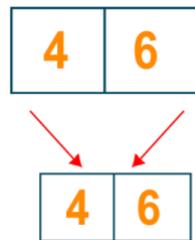
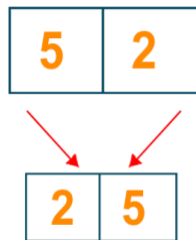
We divide it into two halves as given below



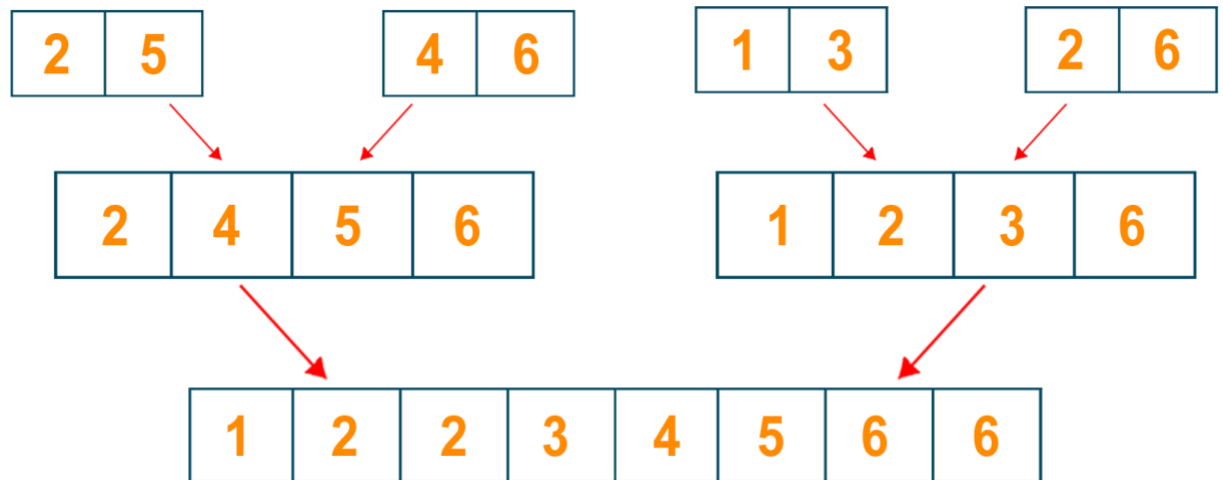
We keep dividing it until the subpart of the list consists of only one element



Later, we will combine the two elements in the given sorted manner



Again, we will combine the two lists of two elements and convert them into a list of four elements after sorting. We will keep repeating the process until we find the final sorted sequence of the array



The final sorted sequence of data is as given



## Time and Space Complexity

The running time complexity for best case, worst case, and average-case scenario is  $O(n \cdot \log n)$  where  $n$  is the number of elements to be sorted. The space complexity of the algorithm is  $O(n)$  where we need an array of size  $n$  to place the sorted element.

## Advantages Of Merge Sort

- Merge sort allows sorting large data set easily
- Merge sort can access data in sequence, hence the need for random access is low
- Merge sort is a stable sorting algorithm
- It has the same running time complexity irrespective of case scenarios

## Disadvantages Of Merge Sort

- It requires an array of the same size as the original list to store the final sorted array which consumes the large memory space
- It is slower when it comes to sorting smaller data sets

## Applications

- Merge sort is used in e-commerce application
- Merge sort is the best way to sort the linked list in comparison to another sorting algorithm
- Merge sort is used in internal and external sorting
- It is used in organizing MP3 library
- It is used to display Google PageRank results
- Merge sort is widely used in inversion count problem