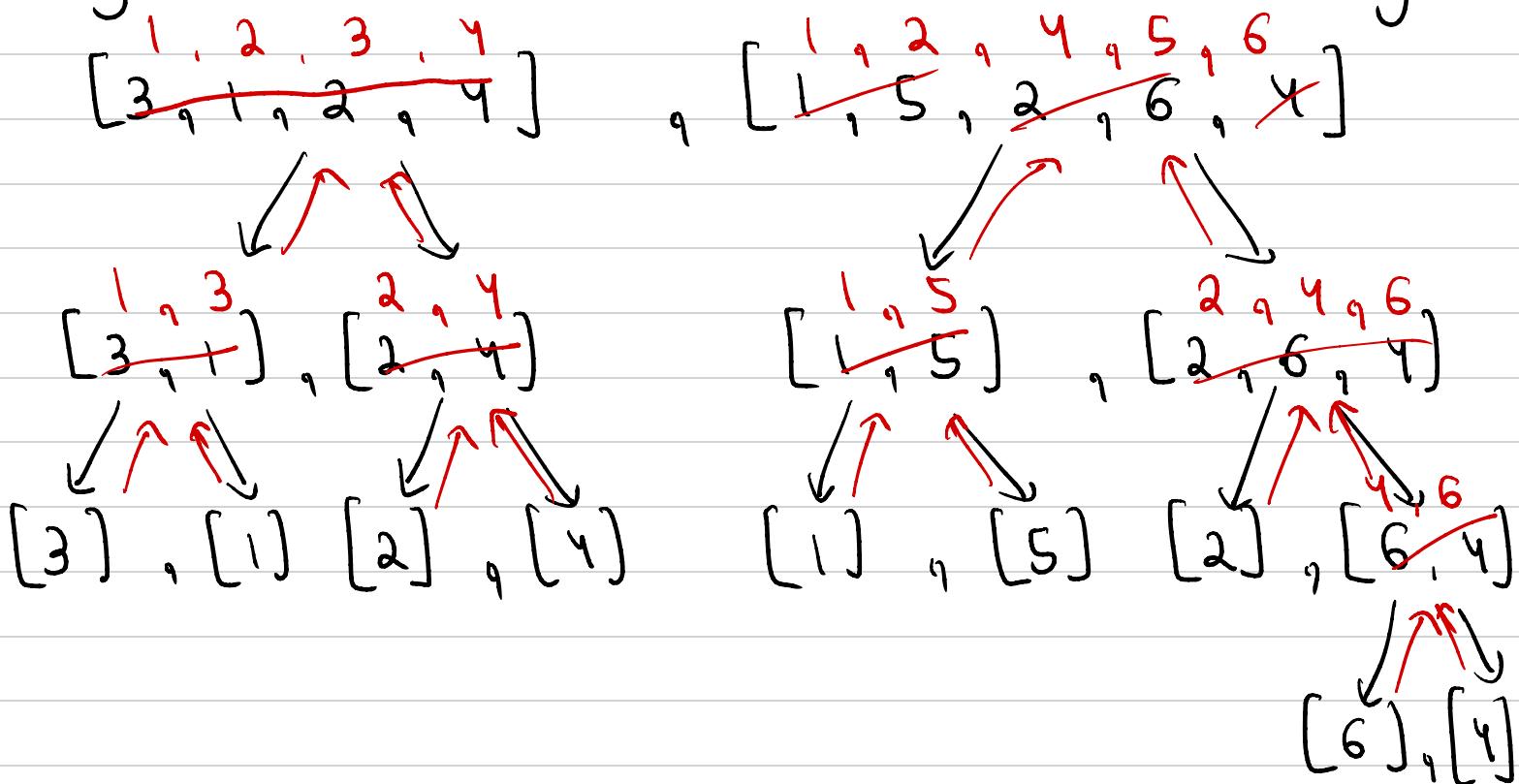


# MERGE SORT

Recursive Sorting Algorithm  
 $O(n \log n)$  time complexity

[3, 1, 2, 4, 1, 5, 2, 6, 4]

Algorithm follows divide & merge



All of this happens hypothetically

Final merge is performed at the end. This is basically dividing the elements into single element arrays and merging them at the end using 2 pointer approach

Pseudocode :

```
mergeSort (arr, low, high) {
```

```

if (low == high) {
    return
}
mid = (low + high) / 2
mergeSort(arr, low, mid)
mergeSort(arr, mid + 1, high)
merge(arr, low, mid, high)
}

```

Dry Run :

0	1	2	3	4
3	2	4	1	3

The merge function is just merging 2 sorted arrays using 2 pointers

