MERGE SORT

Merge Sort is one of the most popular sorting algorithms that is based on the principle of Divide and Conquer Algorithm.

- Here, a problem is divided into multiple sub-problems. Each sub-problem is solved individually. Finally, sub-problems are combined to form the final solution.

Merge Sort Algorithm

- MergeSort is a recursive sorting procedure that uses at most **O(nlog(n))** comparisons.
- To sort an array of **n** elements, we perform the following steps in sequence:
- If **n < 2** then the array is already sorted.
- Otherwise, **n > 1**, and we perform the following three steps in sequence:
    1. **Sort** the **left half** of the the array using MergeSort.
    2. **Sort** the **right half** of the the array using MergeSort.
    3. **Merge** the sorted left and right halves.

Algorithm

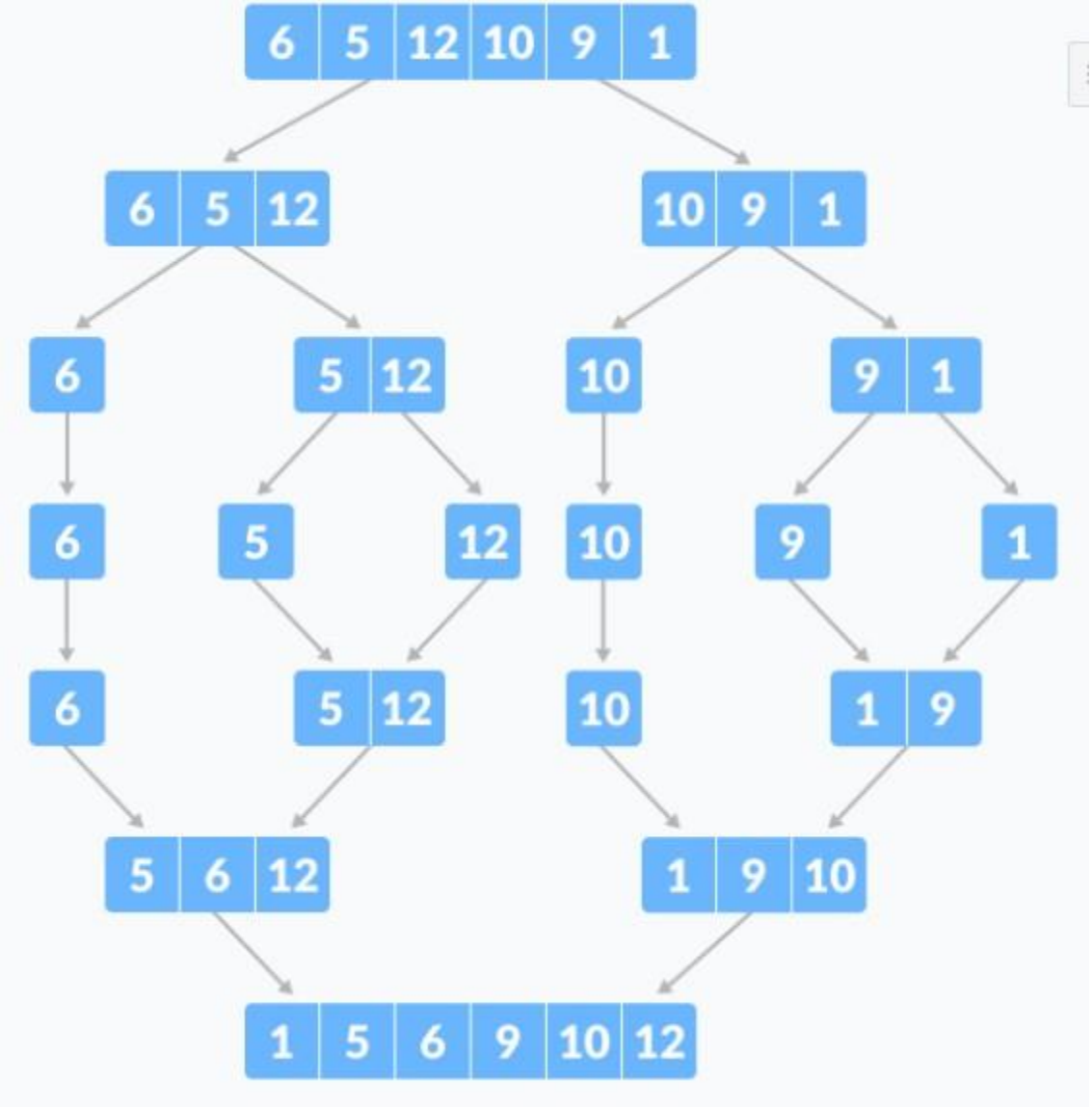
Step 1: Start

Step 2: Declare array and left, right, mid variable

Step 3: Perform merge function.

```
if left > right
return
mid= (left+right)/2 mergesort(array, left, mid)
mergesort (array, mid+1, right)
merge(array, left, mid, right)
```

Step 4: Stop



Program

```
def merge_sort(arr):
 # Base case: if the array has one or zero elements, it's already sorted
if len(arr) <= 1:
 return arr
 # Divide: Find the middle point to divide the array into two halves
 mid = len(arr) // 2
 # Recursively sort the left half and right half
 left = merge_sort(arr[:mid])
```

```
        right = merge_sort(arr[mid:])
        # Combine: Merge the sorted halves
        return merge(left, right)
def merge(left, right):
    sorted_array = []
    i = j = 0
    # Compare elements from both halves and merge them in sorted order
    while i < len(left) and j < len(right):
        if left[i] < right[j]:
            sorted_array.append(left[i])
            i += 1
        else:
            sorted_array.append(right[j])
            j += 1
# If there are any remaining elements in the left half, add them
    while i < len(left):
        sorted_array.append(left[i])
        i += 1
    # If there are any remaining elements in the right half, add them
    while j < len(right):
        sorted_array.append(right[j])
        j += 1
    return sorted_array
```