Difference between Merge Sort and Quick Sort

Merge Sort and Quick Sort are two popular algorithms for sorting an array of elements. Both algorithms have their unique advantages and disadvantages. Here are the main differences between Merge Sort and Quick Sort.

1. Approach

Quick Sort is a partition-based sorting algorithm that works by selecting a pivot element and partitioning the array into two sub-arrays, one with elements smaller than the pivot and another with elements larger than the pivot. The pivot element is then placed in its final position in the sorted array. This process is repeated recursively on the sub-arrays until the entire array is sorted.

Merge Sort is a divide-and-conquer algorithm that works by dividing the array into two halves, sorting each half recursively using Merge Sort, and then merging the two sorted halves to produce the final sorted array.

2. Time Complexity

The time complexity of Quick Sort depends on the choice of pivot element. In the worst case, when the pivot element is the smallest or largest element in the array, Quick Sort takes O(n^2) time. However, on average, Quick Sort takes O(n log n) time.

Merge Sort, on the other hand, has a time complexity of O(n log n) in all cases, regardless of the input distribution.

3. Space Complexity

Quick Sort is an in-place sorting algorithm, meaning it does not require additional memory to sort an array. However, it uses a stack to keep track of the recursive calls, which can take up O(log n) space.

Merge Sort, on the other hand, requires additional memory to store the sorted sub-arrays during the merge step. This makes Merge Sort less space-efficient than Quick Sort.

4. Stability

A sorting algorithm is said to be stable if it maintains the relative order of equal elements in the input array. Merge Sort is a stable sorting algorithm, meaning it preserves the order of equal elements in the array. Quick Sort, on the other hand, is not a stable sorting algorithm.

5. Best Use Case

Quick Sort is faster than Merge Sort in most cases, especially when the input array is large and has a random distribution of elements. This makes Quick Sort a good choice for general-purpose sorting.

Merge Sort, on the other hand, is a good choice when stability is required or when the input array is too large to fit in memory. It is also useful in distributed computing, where the sorting can be parallelized across multiple machines.

In summary, Quick Sort is faster than Merge Sort in most cases, but Merge Sort is more space-efficient and stable. The choice between the two algorithms depends on the specific requirements of the problem at hand.