# Assignment 1

# Muhammad Junaid

# 204-BSCS-22f

# 

Q-1. Behavior of Merge Sort on an Already Sorted Array

Merge Sort, a well-known divide-and-conquer sorting algorithm, operates by recursively dividing the array into two halves, sorting each part, and merging them into a sorted array. When this algorithm is used on an already sorted list, it does not recognize the pre-existing order. The process of splitting and merging is carried out in the same way as it would be for an unsorted array. As a result, the number of comparisons and recursive steps remains unchanged. Unlike algorithms such as Insertion Sort that perform faster on sorted inputs, Merge Sort retains its O(n log n) time complexity regardless of the input’s order whether it’s sorted, reversed, or random. Thus, no optimization occurs in the number of comparisons, making the algorithm reliable but not particularly efficient for sorted data.

Q-2. Efficiency of Merge Sort on Arrays with Many Duplicate Values

Merge Sort works just as well when there are many repeated values in the array. It treats duplicates just like other values and compares them normally during the merging step. If two values are the same, Merge Sort always keeps the one from the left side first. This means it keeps the original order of repeated items, which is called stability. Even though having duplicates doesn't change the overall time needed (still O(n log n)), there may be a small benefit in how much data is moved around. Overall, Merge Sort is a good choice for sorting lists with many repeated items, like database entries.

Q-3. Recursion and Memory Usage in Merge Sort on Systems with Limited Stack Size

Merge Sort uses recursion, which means it keeps calling itself to break the array into smaller parts. But each of these calls uses memory on the system’s stack. The number of these calls grows with the size of the array. For very large arrays, on systems with low memory, this can cause problems like stack overflow. Also, Merge Sort needs extra space to hold temporary arrays while merging, which makes it use more memory (O(n)) compared to sorting methods like Quick Sort or Heap Sort that don’t need as much space. Still, most modern computers can handle this without any issue. The simple logic of recursion makes Merge Sort easy to understand and use.

Q-4. Stable vs. Unstable Sorting Algorithms and Merge Sort's Classification

Sorting methods can be *stable* or *unstable*. A stable sort keeps the original order of equal items, while an unstable one might not. Merge Sort is a stable sorting algorithm. When two equal values are compared during merging, Merge Sort always places the one from the left first, keeping their original order. This is helpful in cases where we sort by multiple things like sorting employees first by department and then by salary. If the sort is stable, the department order stays the same. Other sorts like Quick Sort and Heap Sort are usually unstable. So, Merge Sort is called a stable algorithm and is useful when keeping the order of equal values matters like in spreadsheets or database sorting.