Explain different sorting algorithms (Bubble Sort, Insertion Sort, Quick Sort, Merge Sort).

Bubble Sort

Bubble Sort is a simple comparison-based algorithm that repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the wrong order. This process is repeated until the list is sorted.

Insertion Sort

Insertion Sort builds the final sorted array one item at a time. It is much less efficient on large lists.

Quick Sort

Quick Sort is a highly efficient sorting algorithm and is based on the divide-and-conquer approach. It works by selecting a 'pivot' element from the array and partitioning the other elements into two sub-arrays, according to whether they are less than or greater than the pivot.

Merge Sort

Merge Sort is an efficient, stable, comparison-based, divide-and-conquer sorting algorithm. Most implementations produce a stable sort, meaning that the implementation preserves the input order of equal elements in the sorted output.

Compare the performance (time complexity) of Bubble Sort and Quick Sort.

- **Bubble Sort**: $O(n^2)$
- **Quick Sort**: $O(n \log n)$

Discuss why Quick Sort is generally preferred over Bubble Sort.

Quick Sort is generally preferred over Bubble Sort because it has a significantly better average-case time complexity of $O(n \log n)$ compared to Bubble Sort's $O(n^2)$. This makes Quick Sort much more efficient for large datasets. Additionally, Quick Sort is an in-place sorting algorithm, meaning it requires only a small, constant amount of additional storage space.