

Question: *Explain the QuickSort algorithm with an example.*

Teacher's Solution:

QuickSort is a **divide-and-conquer** algorithm that sorts an array by selecting a **pivot** element, partitioning the array into two subarrays (one with elements smaller than the pivot and one with elements greater), and recursively sorting the subarrays.

Steps of QuickSort:

1. Choose a pivot element.
2. Partition the array such that elements smaller than the pivot are on the left and larger elements are on the right.
3. Recursively apply QuickSort to the left and right subarrays.

Example: Sorting [8, 4, 7, 3, 9]

- Choose 7 as pivot.
- Partition: [4, 3] (pivot) [8, 9]
- Recursively sort [4, 3] → [3, 4] and [8, 9] → [8, 9].
- Final sorted array: [3, 4, 7, 8, 9].

Time Complexity:

- **Best & Average Case:** $O(n \log n)$
- **Worst Case (Already sorted in reverse order):** $O(n^2)$

Advantages: Efficient for large datasets.

Disadvantages: Worst-case time complexity can be slow if not implemented with a good pivot strategy.