CO-1

ALM

Q) How do we compare two algorithms that solve the same problem but have differant time and space complexities?

1) Analyzing time complexity:-

- Asymptotic Notations:- Determine O, Θ, Ω notations both algorithms to understand worst-case, best-case, Average case.

- Input size:- Consider how each algorithm performs as input size increases. An algorithm with a better time complexity may outperform the other as input size grows larger.

2) Analyzing space complexity:-

- Memory usage:- Determine space complexity for each algorithm. this includes not just extra space used but also space required to store input & output.

Trade-offs:- Analyze how space requirements grow with input size. An algorithm with better space might be more suitable for memory constrained environments.

→ lets compare two sorting algorithms:-

a) Merge sort & Insertion sort.

→ Merge sort:-

Time complexity:-

worst-case $O(n \log n)$

Average-case $O(n \log n)$

Best-case :- $O(n \log n)$

space complexity:- $O(n)$

→ Insertion sort:-

Time complexity:-

worst case:- $O(n^2)$

Average case:- $O(n^2)$

Best case:- $O(n)$

Space complexity:- $O(1)$