Data Structures & Algorithms Using Java (DSA using Java)

1. Complexity Analysis

- Big-O Notation
- Time & Space Complexity
- Best, Average, Worst Case Analysis

2. Arrays

- Array operations (insert, delete, search)
- Prefix Sum Technique
- Two Pointer Approach
- Sliding Window

Problems:

- Maximum Subarray (Kadane's Algorithm)
- Two Sum / Pair Sum
- Move Zeroes
- Merge Intervals
- Stock Buy & Sell
- Subarray with given sum

3. Strings

- String operations & manipulation
- Palindrome problems
- Anagram problems
- Substring problems

4. Linked List

- Singly Linked List
- Doubly Linked List (basics)

Problems:

- Reverse Linked List
- Detect Cycle (Floyd Algorithm)
- Middle of Linked List
- Merge Two Linked Lists

5. Stack & Queue

- Stack implementation
- Queue implementation
- Priority Queue (Heap concept)

Problems:

- Valid Parentheses
- Next Greater Element
- Sliding Window Maximum

6. Searching & Sorting

- Linear Search
- Binary Search (important)
- Binary Search variations
- Sorting:
    - Bubble Sort
    - Selection Sort
    - Insertion Sort
    - Merge Sort
    - Quick Sort (concept)

7. Recursion & Backtracking

- Recursion basics
- Base case & recursive calls

Problems:

- Factorial & Fibonacci
- Subsets
- Permutations
- N-Queens

8. Trees

- Binary Tree
- Binary Search Tree

Concepts:

- Tree Traversals (Inorder, Preorder, Postorder)
- Level Order Traversal

Problems:

- Height of Tree
- Diameter of Tree
- Lowest Common Ancestor
- Count Nodes

9. Heap (Priority Queue)

- Min Heap & Max Heap
- Heap operations

Problems:

- Kth Largest Element
- Top K Elements

10. Hashing

- HashMap & HashSet
- Frequency counting
- Collision handling (basic idea)

Problems:

- Two Sum (optimized)
- First non-repeating element

11. Dynamic Programming

- Overlapping subproblems
- Memoization & Tabulation

Problems:

- Fibonacci (DP)
- Climbing Stairs
- 0/1 Knapsack (basic)
- Longest Common Subsequence

12. Graphs

- Graph representation
- BFS & DFS

Problems:

- Number of Islands
- Shortest Path (basic idea)