■ Tree Data Structures Roadmap

1. Basics of Trees

- Definitions: Root, Node, Leaf, Degree, Depth, Height, Subtree.
- Tree Terminology: Ancestor, Descendant, Sibling, Path, etc.
- Types of Trees: Binary Tree, N-ary Tree, Full Tree, Complete Tree, Balanced Tree.

2. Binary Trees

- Binary Tree Representation: Using arrays and linked structures.
- Tree Traversal Methods:
- • Preorder (Root-Left-Right)
- • Inorder (Left-Root-Right)
- • Postorder (Left-Right-Root)
- • Level Order Traversal (BFS)
- Implement both recursive and iterative versions.

3. Binary Search Trees (BST)

- BST Property: Left < Root < Right.
- Basic Operations: Insertion, Deletion, Search.
- Important BST Problems:
- • Validate BST
- • Find Minimum and Maximum
- • Floor and Ceil in BST

4. Advanced Binary Trees

- Balanced Trees: Intro to AVL Trees and Red-Black Trees (conceptual).
- Segment Trees (for range queries) – only overview required.
- Threaded Binary Tree – optional awareness.

5. Key Tree Problems (for practice)

- • Convert Sorted Array to BST – (LeetCode #108)
- • Lowest Common Ancestor – (LeetCode #235, #236)
- • Diameter of Binary Tree – (LeetCode #543)
- • Zigzag Level Order Traversal – (LeetCode #103)
- • Boundary Traversal of a Binary Tree
- • All Paths From Root to Leaf – (LeetCode #257)
- • Invert/Flip Binary Tree – (LeetCode #226)