1. Find Height of a Binary Tree

Find the maximum depth (height) of a tree using recursion.

Example:

Input:

Output: 3

2. Count Leaf and Non-Leaf Nodes

Count how many nodes are **leaves** (no children) and how many are **non-leaves.**

4. Level Order Traversal

Traverse the tree level by level using a queue.

Example:

```
 Input:

   1

   / \

 2    3


Output: 1 2 3
```

5. Diameter of a Binary Tree (Leetcode #543)

Find the **longest path** between any two nodes in a tree.

6. Check if a Binary Tree is a BST

Verify if all nodes satisfy BST properties.

7. Find Path from Root to a Given Node

Given a node value, print the path from root to that node.

8. Sum of All Left Leaves

Find the sum of all **left leaf nodes** in a binary tree.

9. Print Ancestors of a Given Node

Given a node x, print all its **ancestors** from root to parent.

Example:

Input:

```
      1
     / \
    2   3
   / \
  4   5
```

Ancestors of 5: 2, 1

10. Find Kth Smallest Element in a BST

Return the **Kth smallest** element using inorder traversal.

Example:

Input: `[3, 1, 4, null, 2]`, `k = 1`

Output: `1`