

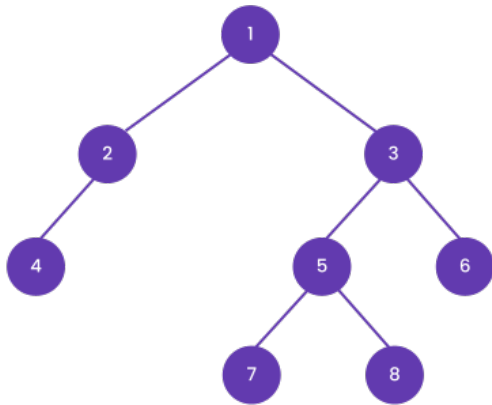
# Binary tree

## Assignment Questions



**Q1. Given the root of a binary tree, return the spiral level order traversal of its nodes' values. The solution should consider the binary tree nodes level by level in spiral order, i.e., all nodes present at level 1 should be processed first from left to right, followed by nodes of level 2 from right to left, followed by nodes of level 3 from left to right and so on... In other words, odd levels should be processed from left to right, and even levels should be processed from right to left.**

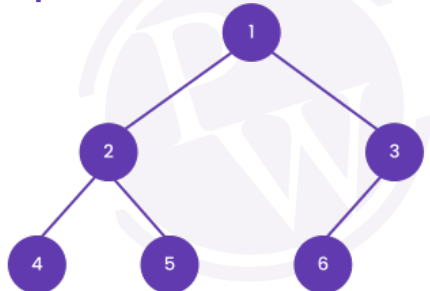
**Input:**



**Output:** [1, 3, 2, 4, 5, 6, 8, 7]

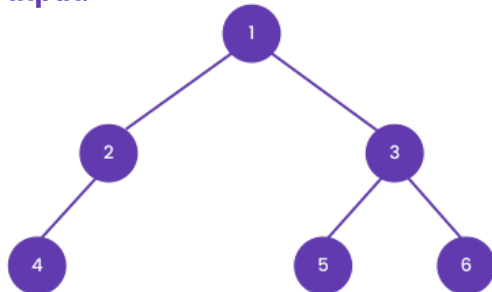
**Q2. Given the root of a binary tree, check if it is a complete binary tree or not. A complete binary tree is a binary tree in which every level, except possibly the last, is filled, and all nodes are as far left as possible.**

**Input:**



**Output:** Complete Binary Tree

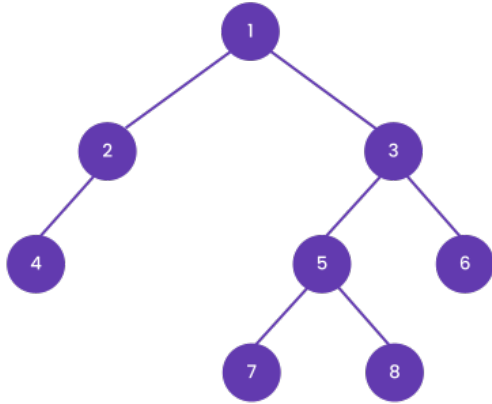
**Input:**



**Output:** Not a Complete Binary Tree

**Q3. Given the root of a binary tree, return the reverse level order traversal of its nodes' values. The solution should consider the binary tree nodes level by level in bottom-up order from left to right, i.e., process all nodes of the last level first, followed by all nodes of the second last level, and so on.**

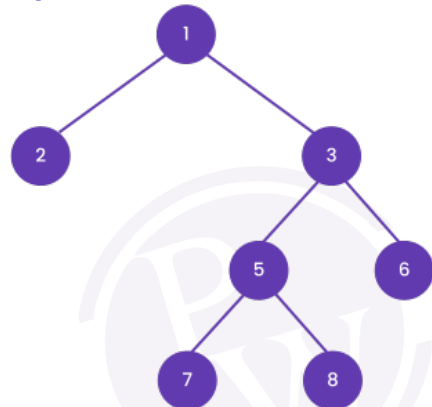
Input:



Output: [7, 8, 4, 5, 6, 2, 3, 1]  
\* /

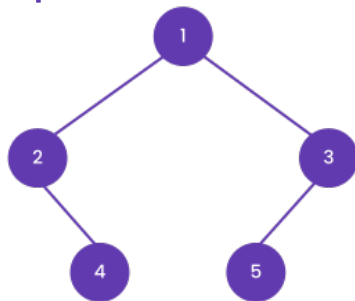
Q4. Given the root of a binary tree, return the left view of its nodes' values. Assume the left and right child of a node makes a 45-degree angle with the parent.

Input:



Output: [1, 2, 5, 7]

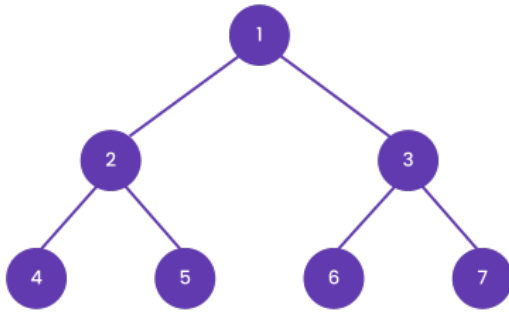
Input:



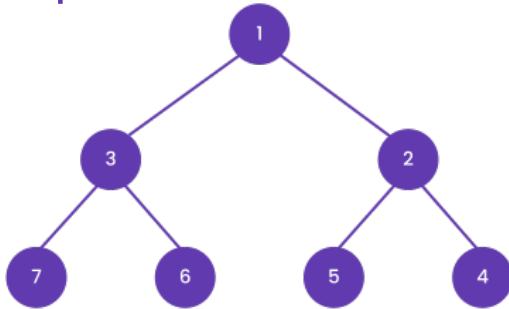
Output: [1, 2, 4]

Q5. Given the root of a binary tree, convert the binary tree into its mirror and print it's pre order

Input:



**Output:**



**Explanation:**

