**Lowest Common Ancestor (LCA) in a Binary Tree**

You are given a binary tree and two distinct nodes within the tree. Your task is to find the lowest common ancestor (LCA) of these two nodes. The LCA of two nodes p and q is defined as the lowest node in the tree that has both p and q as descendants (where we allow a node to be a descendant of itself).

**Input:**

* A binary tree represented as a series of nodes where each node has references to its left and right child.
* Two distinct nodes p and q from the tree.

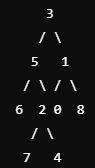
**Output:**

* Return the node that is the lowest common ancestor (LCA) of p and q.

**Examples:**

* Example 1  
  Input: root = [3, 5, 1, 6, 2, 0, 8, null, null, 7, 4], p = 5, q = 1

Output: 3  
Explanation:

* The tree is as follows  
  
* The lowest common ancestor of nodes 5 and 1 is 3, because 3 is the lowest node that has both 5 and 1 as descendants.

**Constraints:**

* The number of nodes in the tree is between 2 and 105
* Each node’s value is unique.
* Both p and q are distinct and guaranteed to be in the tree.

**Test Cases:**

1. Input: root = [3, 5, 1, 6, 2, 0, 8, null, null, 7, 4], p = 5, q = 4

Output: 5

1. Input: root = [3, 5, 1, 6, 2, 0, 8, null, null, 7, 4], p = 5, q = 1

Output: 3

1. Input: root = [1, 2], p = 1, q = 2

Output: 1

**Edge Cases:**

1. Both Nodes are the Same: If p and q are the same node, the LCA is the node itself
2. One Node is Root: If one node is the root of the tree, the root is the LCA.
3. Small Tree: The tree has only two nodes.