Populating Next Right Pointers in Each Node II

This problem differs from the perfect binary tree version because the tree may not be complete.

Key Idea:

- We must connect nodes level by level.
- Each next pointer should link to the adjacent node on the same level.
- Constraints: Constant extra space → no queue (BFS) allowed, recursion stack is allowed.

Approach: Iterative constant space (Two pointer):

We use three pointers:

1. curr: iterates over the current level.
2. head: first node of the next level.
3. prev: tracks the previous node in the next level to set its next.

Steps:

1. Initialize curr = root.
2. For each level:
   - Traverse nodes using curr → next.
   - Connect children (left, right) for the next level:
     - If prev is not nullptr, link prev → next to child.
     - Otherwise, set head to this child (start of next level).
   - Move prev to this child.
3. After finishing one level, move curr = head (start next level) and reset head and prev to nullptr.

Complexity:

- Time: O(N) – each node visited once.
- Space: O(1) extra (only pointers, no additional data structures; recursion

not used here).