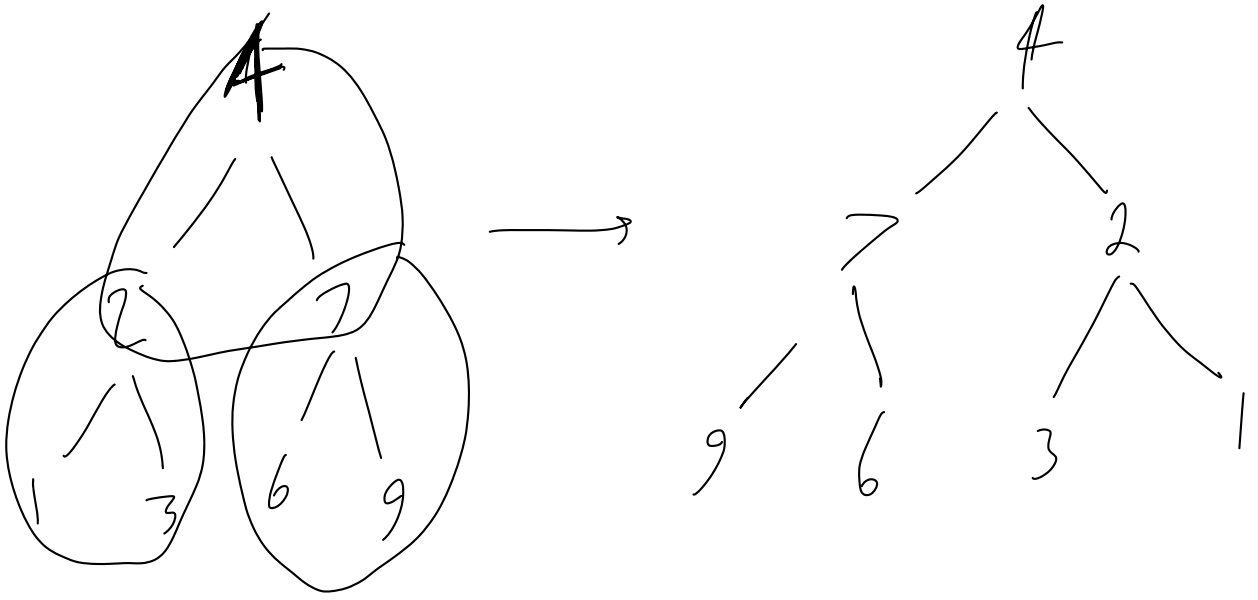
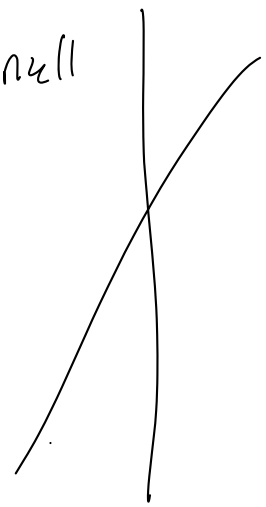
Given the root of a binary tree, invert the tree and return its root.



I think we need to create some dummy node.

```
while dummy.left and dummy.right is not null
    another = new TreeNode()
    another.val = dummy.right
    dummy.right = dummy.left

    dummy.left = another.val
```



Can use recursion

DFS → Depth First Search

```
if not root:
    return None
```

```
dummy = root.left
root.left = dummy
root.right = dummy
```

time → O(N)

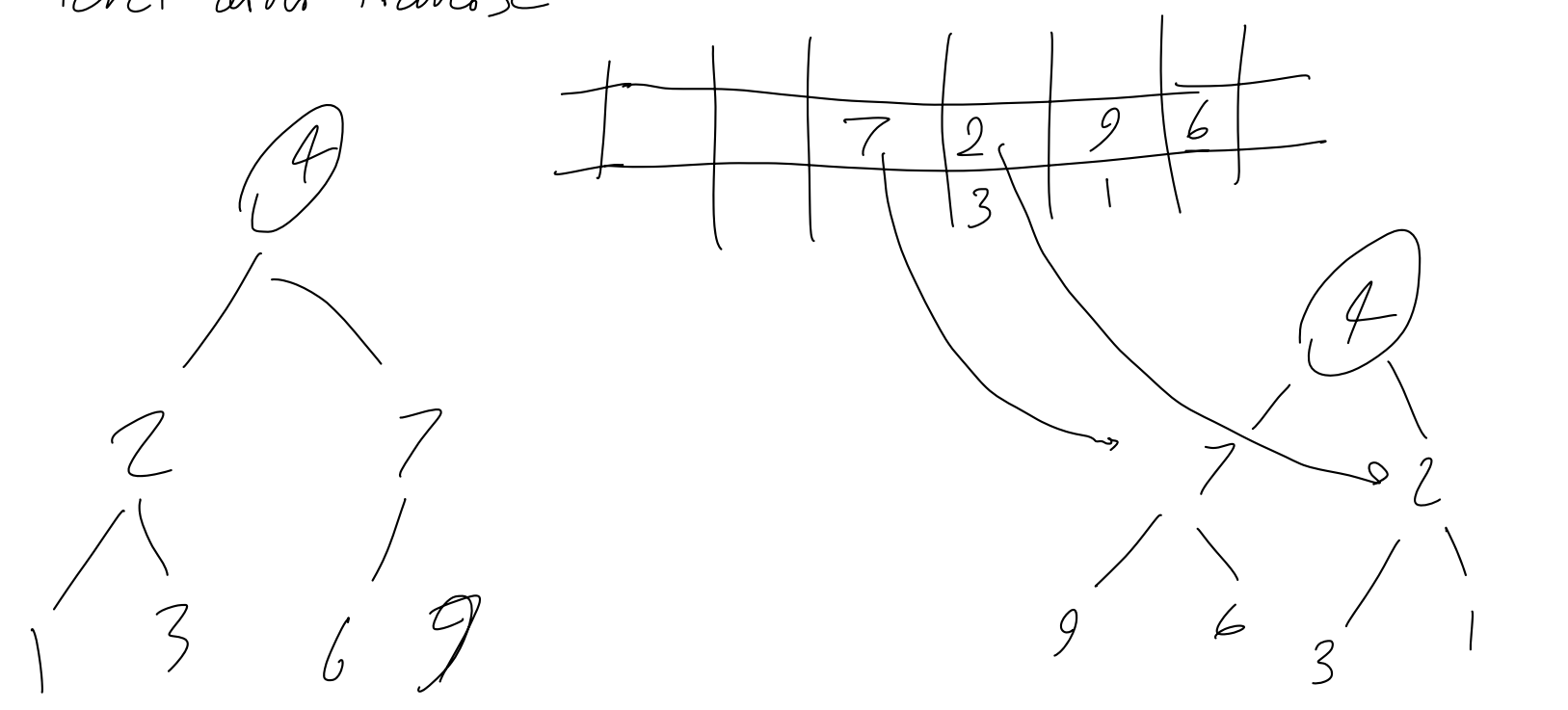
space → O(depth)

↳ better than O(N)

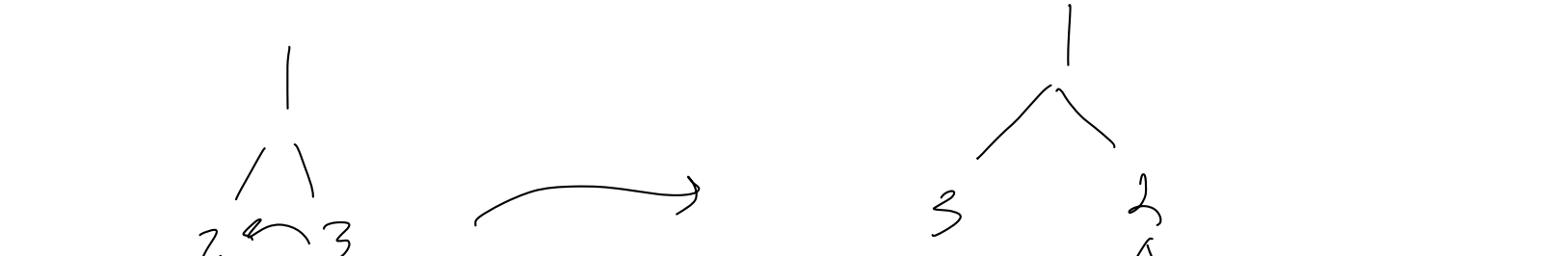
```
self.invert(root.left)
self.invert(root.right)
```

---

level order Traverse

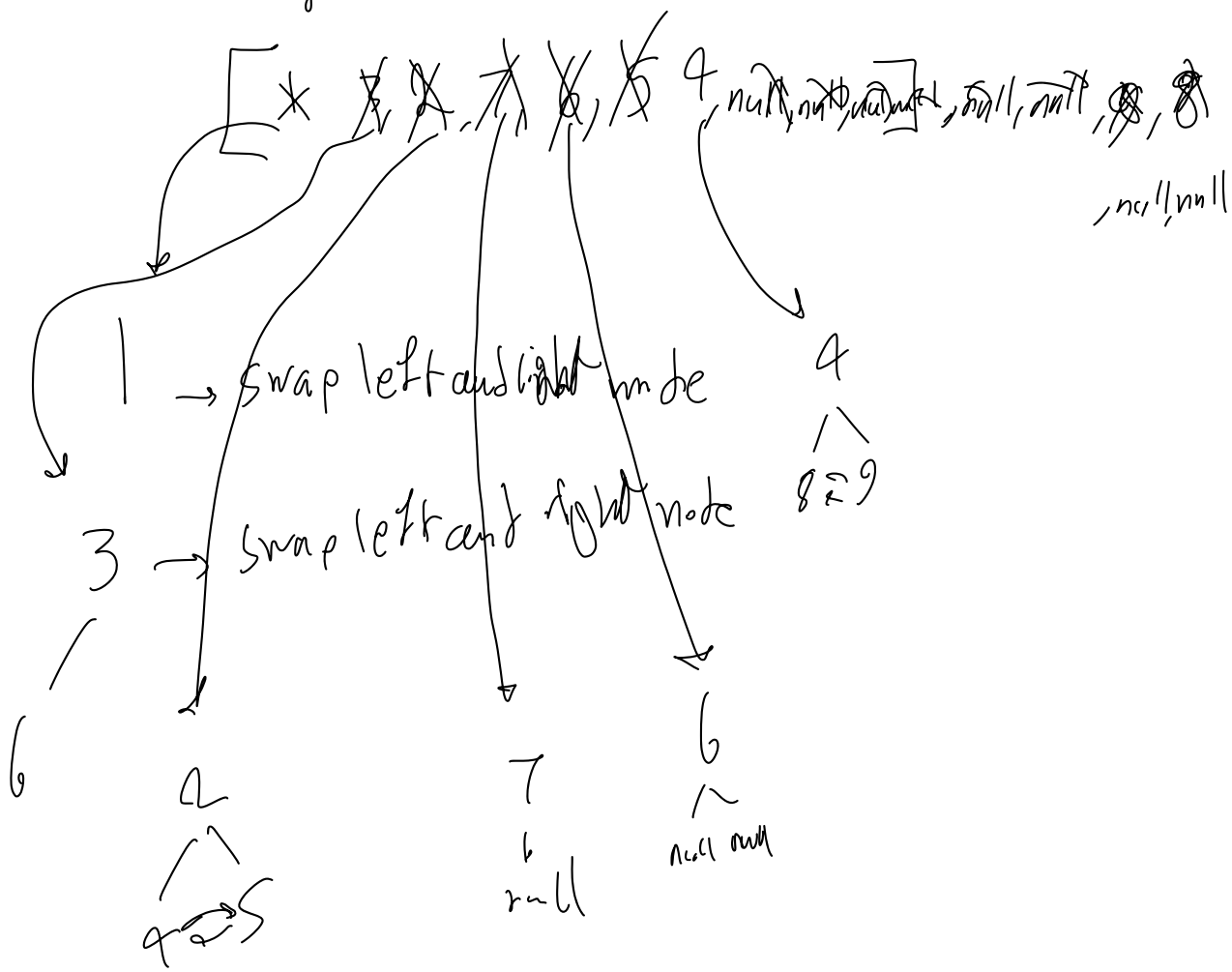


AlgoExpert Soln



Ittereative → Check out BFS

initialize a queue



Time → O(n)

Space → O(n)