Invert Binary Tree

Try to solve the Invert Binary Tree problem.

We'll cover the following

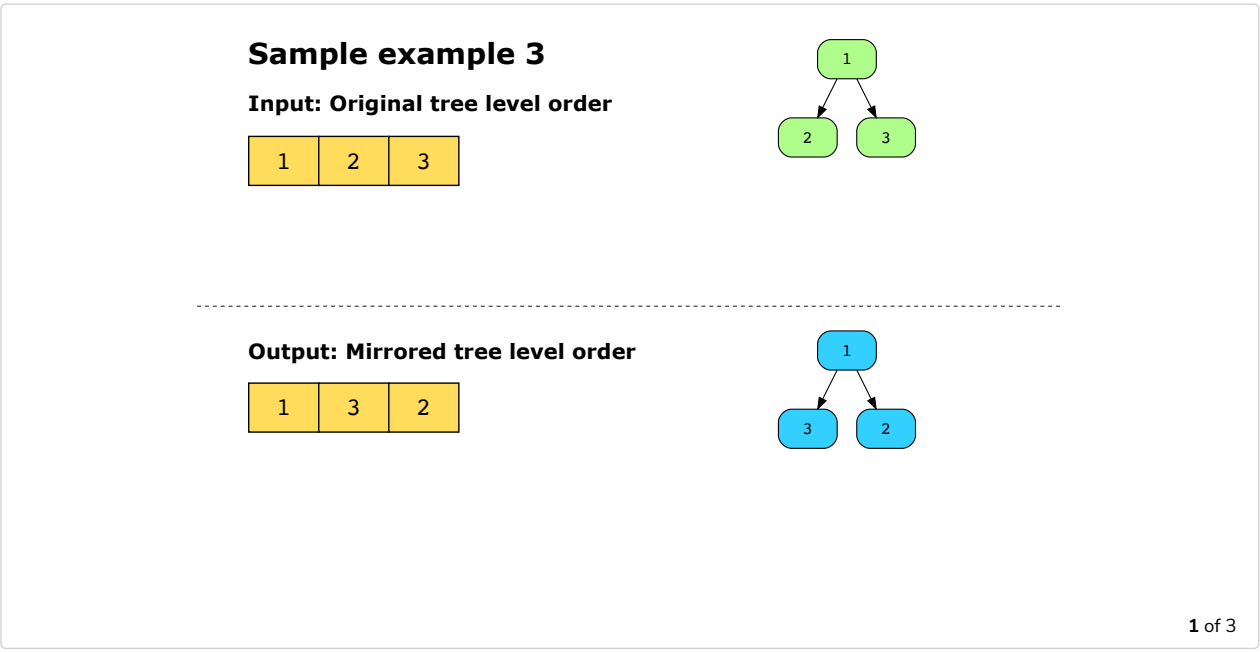
Statement

Given the root node of a binary tree, convert the binary tree into its mirror image, and return the root of the converted tree.

Constraints:

- $0 \leq$ Number of nodes in the tree ≤ 100
- $-1000 \leq$ `Node.value` ≤ 1000

Examples



Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Invert Binary Tree

1 What is the correct mirrored tree for the given tree?

A)

B)

C)

D)

Submit Answer

Question 1 of 2
0 attempted

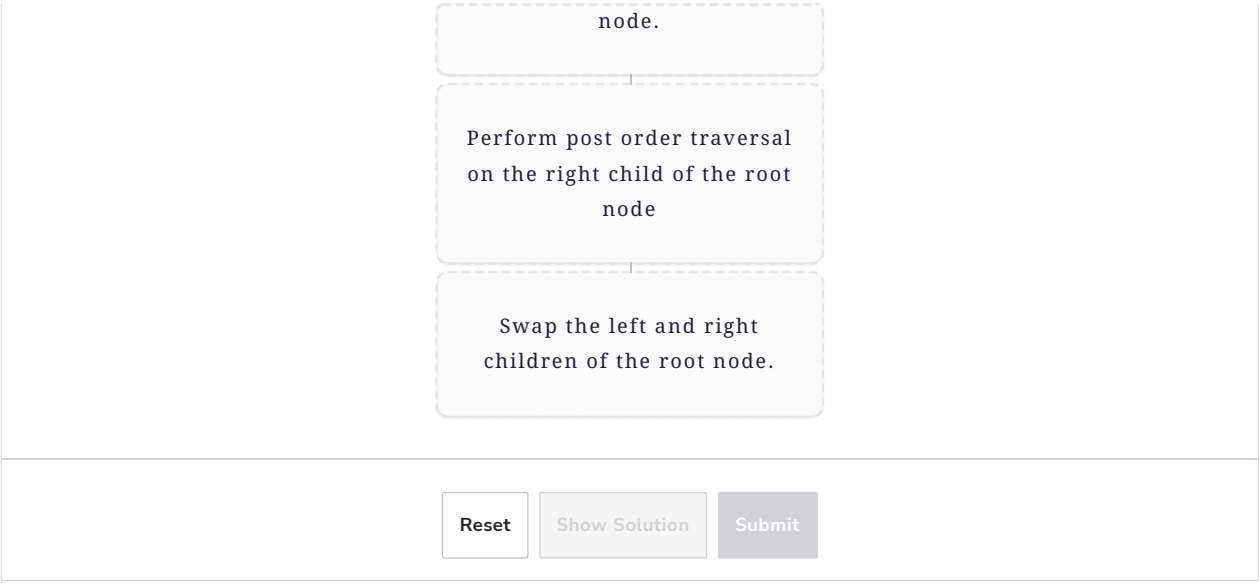
Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Perform post order traversal on the left child of the root



Reset | Show Solution | Submit

Try it yourself

Implement your solution in `main.java` in the following coding playground.

Java

usercode > main.java

```
// Definiton of a binary tree node class
// class TreeNode<T> {
//     T data;
//     TreeNode<T> left:
//     TreeNode(T data) {
//         this.data = data;
//         this.left = null;
//         this.right = null;
//     }
// }

import java.util.*;
import ds_v1.BinaryTree.TreeNode;

public class Main{

    public static TreeNode<Integer> invertTree(TreeNode<Integer> root){

        // Replace this placeholder return statement with your code
        return root;
    }
}
```

Powered by AI | Submit

Test Cases | Results

Case 1 | Case 2 | Case 3

Input #1

[100,50,200,25,75,125,350]

Invert Binary Tree