Description | Editorial | Solutions (7.9K) | Submissions

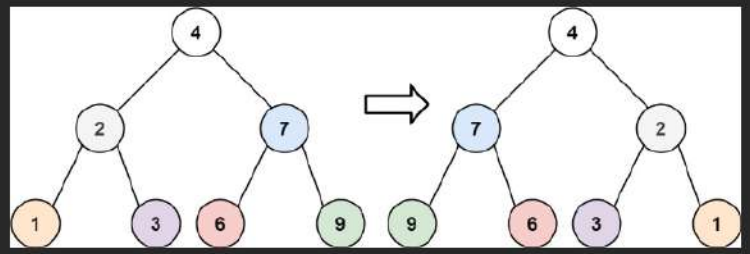
226. Invert Binary Tree

Easy | 12.4K | 175

Companies

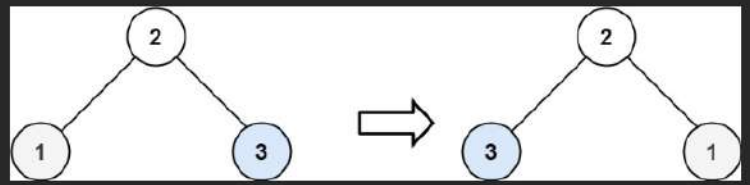
Given the `root` of a binary tree, invert the tree, and return *its root*.

Example 1:




```
Input: root = [4,2,7,1,3,6,9]
Output: [4,7,2,9,6,3,1]
```

Example 2:




```
Input: root = [2,1,3]
Output: [2,3,1]
```

Example 3:

Java | Auto

```java
/**
 * Definition for a binary tree node.
 * public class TreeNode {
 *     int val;
 *     TreeNode left;
 *     TreeNode right;
 *     TreeNode() {}
 *     TreeNode(int val) { this.val = val; }
 *     TreeNode(int val, TreeNode left, TreeNode right) {
 *         this.val = val;
 *         this.left = left;
 *         this.right = right;
 *     }
 * }
 */
class Solution {
    public TreeNode invertTree(TreeNode root) {
        if (root != null) {
            TreeNode temp = root.right;
            root.right = root.left;
```

Testcase | Result

Accepted Runtime: 0 ms

Case 1 | Case 2 | Case 3

Input

root =
[4,2,7,1,3,6,9]

Output

[4,7,2,9,6,3,1]

Expected

Console | Run | Submit

Description | Editorial | Solutions (388) | Submissions

956. Tallest Billboard

Hard | 1.9K | 50

Companies

You are installing a billboard and want it to have the largest height. The billboard will have two steel supports, one on each side. Each steel support must be an equal height.

You are given a collection of `rods` that can be welded together. For example, if you have rods of lengths `1`, `2`, and `3`, you can weld them together to make a support of length `6`.

Return *the largest possible height of your billboard installation*. If you cannot support the billboard, return `0`.

Example 1:

```
Input: rods = [1,2,3,6]
Output: 6
Explanation: We have two disjoint subsets {1,2,3} and {6}, which have the same sum = 6.
```

Example 2:

```
Input: rods = [1,2,3,4,5,6]
Output: 10
Explanation: We have two disjoint subsets {2,3,5} and {4,6}, which have the same sum = 10.
```

Example 3:

```
Input: rods = [1,2]
Output: 0
Explanation: The billboard cannot be supported, so we return 0.
```

Java | Auto

```java
class Solution {
    public int tallestBillboard(int[] rods) {
        int sum = 0;
        for (int rod : rods) {
            sum += rod;
        }

        int[] dp = new int[sum + 1];
        Arrays.fill(dp, -1);
        dp[0] = 0;

        for (int rod : rods) {
            int[] dpCopy = dp.clone();

            for (int i = 0; i <= sum - rod; i++) {
                if (dpCopy[i] < 0) continue;

                dp[i + rod] = Math.max(dp[i + rod], dpCopy[i]);
                dp[Math.abs(i - rod)] = Math.max(dp[Math.abs(i - rod)], dpCopy[i] + Math.min(i, rod));
            }
        }
```

Testcase | Result

Accepted Runtime: 0 ms

Case 1 | Case 2 | Case 3

Input

rods =
[1,2,3,6]

Output

6

Expected

Console | Run | Submit