Experiment-2.1

Student Name: Himanshu
UID: 20BCS7944
Branch: BE-CSE
Section/Group: 905/A
Semester: 6th
Date of Performance: 30/03/2023
Subject Name: Competitive Coding-II
Subject Code: 20CSP-351

1. Aim:

To demonstrate the concept of Tree.

2. Objective:

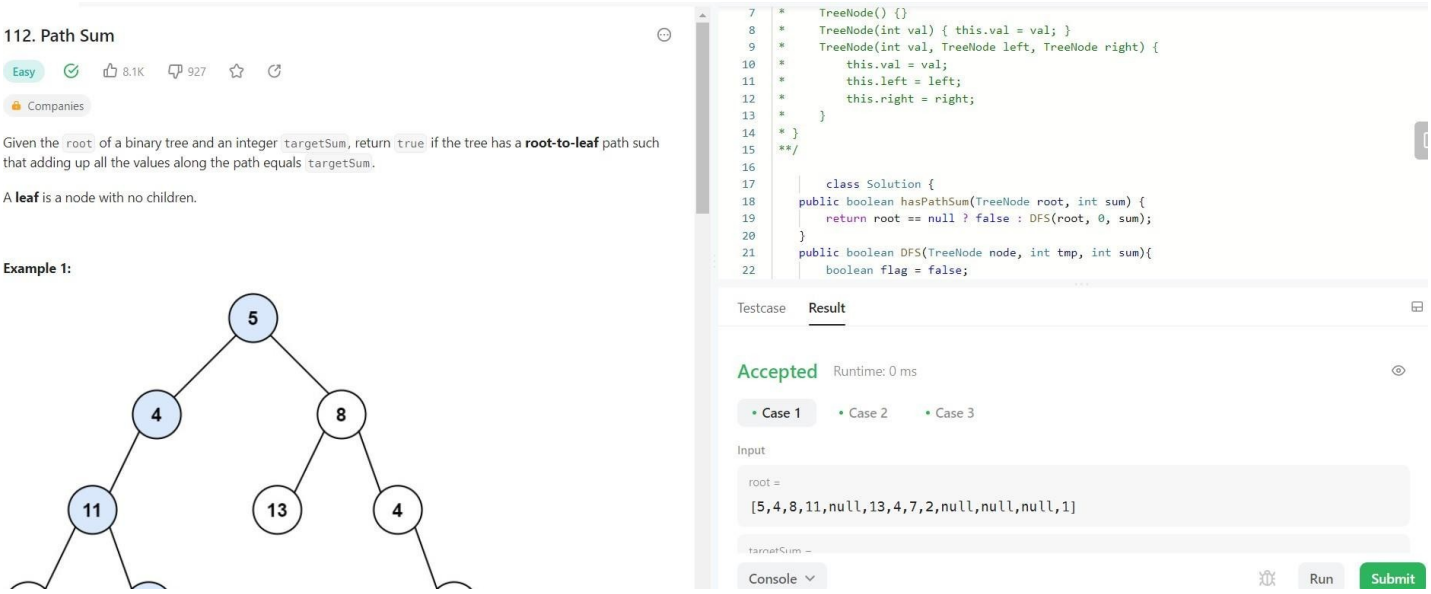
- The objective is to build problem solving capability and to learn the basic concepts of Tree data structures.
- The implementation of Path Sum which shows and brushes up the concept of Tree.
- The implementation of Same Tree.

3. LeetCode code and output:

- **Path Sum -**

```
class Solution {
public boolean hasPathSum(TreeNode root, int sum) {
        return root == null ? false : DFS(root, 0, sum);
}
public boolean DFS(TreeNode node, int tmp, int sum){
        boolean flag = false;
        tmp += node.val;
if(node.left == null && node.right == null){
        return tmp == sum;
}
if(node.left != null){
        flag = flag || DFS(node.left, tmp, sum);
}
if(!flag && node.right != null){
        flag = flag || DFS(node.right, tmp, sum);
}
        return flag;
}
}
```

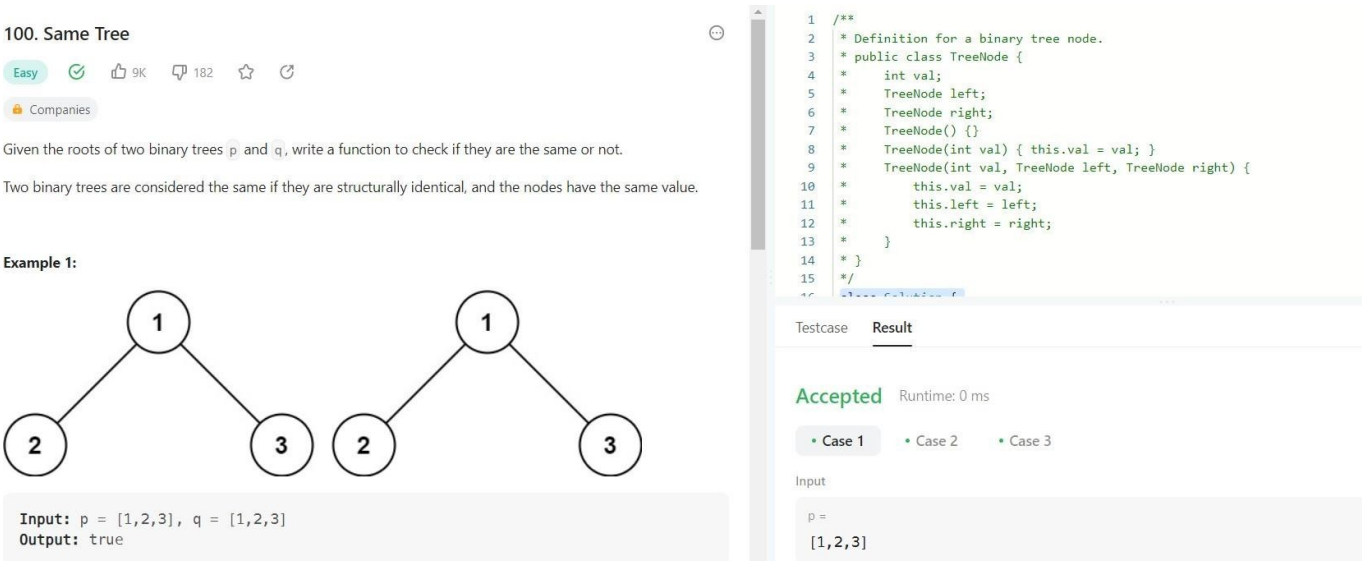
OUTPUT:



- **Same Tree -**

```
class Solution {
    public boolean isSameTree(TreeNode p, TreeNode q) {
if(p==null || q==null)
    return (p==q);
return (p.val==q.val) && isSameTree(p.left,q.left) && isSameTree(p.right,q.right);
  }
}
```

OUTPUT:



100. Same Tree

Easy 9K 182

Companies

Given the roots of two binary trees `p` and `q`, write a function to check if they are the same or not.

Two binary trees are considered the same if they are structurally identical, and the nodes have the same value.

Example 1:


```
Input: p = [1,2,3], q = [1,2,3]
Output: true
```

```
/**
 * 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;
 *     }
 * }
 */
```

Testcase Result

Accepted Runtime: 0 ms

• Case 1 • Case 2 • Case 3

Input

```
p =
[1,2,3]
```