|  |  |
| --- | --- |
| **Experiment-2.1** | |
| **Student Name: Shubham Kumar** | **UID: 21BCS7104** |
| **Branch: CSE** | **Section/Group: 614-A** |
| **Semester: 6th** | **Date of Performance: 07-02-2024** |
| **Subject Name: Advanced Programming Lab-2** | **Subject Code: 21CSP-351** |

**1. Aim:** Trees.

A) Given the roots of two binary trees p and q, write a function to check if they are the same or not.(100. Same tree)

B) Given the root of a binary tree, check whether it is a mirror of itself (i.e., symmetric around its center).(101. Symmetric tree)

**2. Source Code/Output:**

**A) Code:**

class Solution {

public:

bool isSameTree(TreeNode\* p, TreeNode\* q) {

if (p == q) return true;

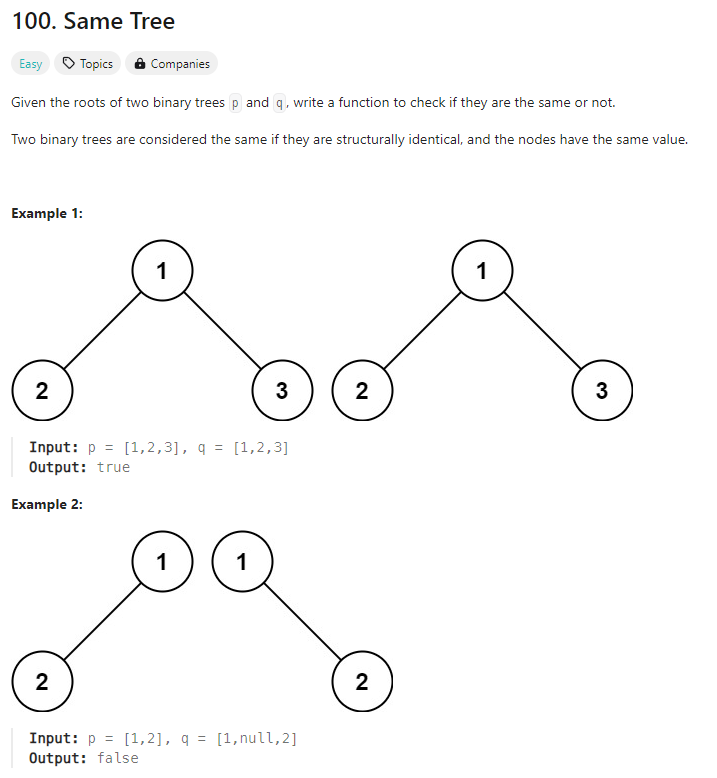
if (!p || !q || p->val != q->val) return false;

return isSameTree(p->left, q->left) && isSameTree(p->right, q->right);

}

};

**Description:**

****

**Output:**



**B)** **Code:**

class Solution {

public:

bool isSymmetric(TreeNode\* root) {

return isSymmetric(root, root);

}

private:

bool isSymmetric(TreeNode\* p, TreeNode\* q) {

if (!p || !q)

return p == q;

return p->val == q->val && //

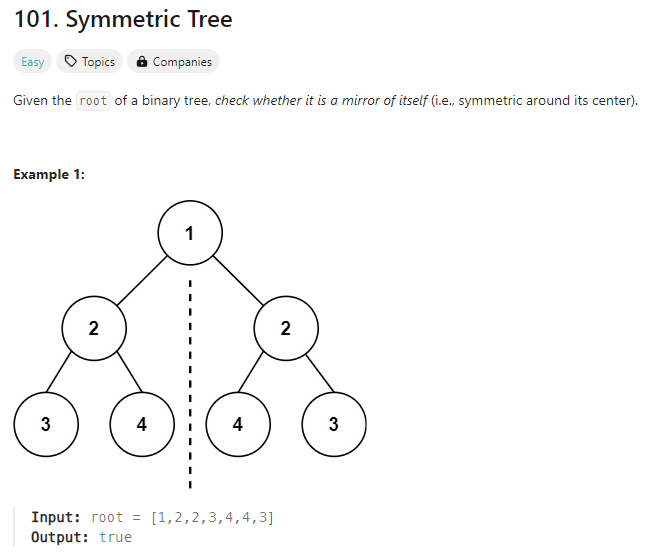
isSymmetric(p->left, q->right) && //

isSymmetric(p->right, q->left);

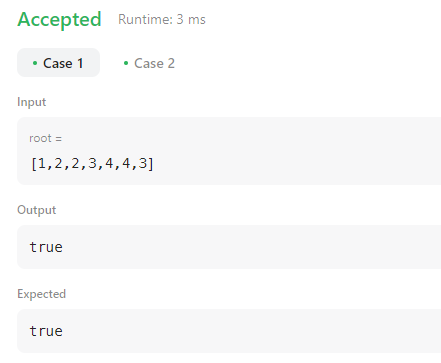
}

};

**Description:**



**Output:**

****