



Experiment 5

Student Name: K.SATWIK

UID 22BCS16246

Branch: CSE

Section/Group: IOT_NTPP_602-A

Semester: 6th

Date of Performance: 18-02-25

Subject Name: AP- 2

Subject Code: 22CSP-351

Aim:

- a) Symmetric tree
- b) Validate binary search tree
- c) Maximum depth of binary tree

Objective: To learn about linked list.

Code:

a)

```
class Solution {
public:
    bool isMirror(TreeNode* t1, TreeNode* t2) {
        if (!t1 && !t2) return true;
        if (!t1 || !t2) return false;
        return (t1->val==t2->val)&&isMirror(t1->right,t2->left)&&isMirror(t1->left,t2->right);
    }
    bool isSymmetric(TreeNode* root) {
        return isMirror(root, root);
    }
};
```

b)

```
class Solution {
public:
    bool isValidBSTHelper(TreeNode* node, long long lower, long long upper) {
        if (!node) return true;
        if (node->val <= lower || node->val >= upper) return false;
```

```
return isValidBSTHelper(node->left, lower, node->val) && isValidBSTHelper(node->right,
node->val, upper);
}
bool isValidBST(TreeNode* root) {
return isValidBSTHelper(root, LLONG_MIN, LLONG_MAX);
}
};
```

```
c)
class Solution {
public:
int maxDepth(TreeNode* root) {
if (!root) return 0;
return 1 + max(maxDepth(root->left),maxDepth(root->right));
}};
```

Output:

a)

☒ Testcase
 | >_ Test Result

Accepted Runtime: 0 ms

☒ Case 1
 ☐ Case 2

Input

```
root =
[1,2,2,3,4,4,3]
```

Output

```
true
```

Expected

```
true
```

b)

☒ Testcase
 | >_ Test Result

Accepted Runtime: 0 ms

☒ Case 1
 ☐ Case 2

Input

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

Output

```
true
```

Expected

```
true
```

c)

☒ Testcase
 | >_ Test Result

Accepted Runtime: 0 ms

☒ Case 1
 ☐ Case 2

Input

```
root =
[3,9,20,null,null,15,7]
```

Output

```
3
```

Expected

```
3
```

Learning Outcomes:

- Understand the concept of trees.
- Learnt about different problem like validate binary search tree, symmetric tree.
- Gain an understanding about the efficiency of tree