# **Format for uploading details in GitHub and Slack in word file format**

**Student Name: Suhas M S**

**Class and Sec: VI B**

**USN: 4AL17CS100**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Online Test Details** | | | | |
| **Subject** |  | | | |
| **Semester** | **VI - B** | | **Duration** | **Minutes** |
|  | |  | | |

**Encl: snapshot of the test result**

|  |  |  |  |
| --- | --- | --- | --- |
| **Pre-Placement Training Summary** | | | |
| **Pre placement training** | **----** | | |
| **Faculty** | **----** | **Duration** | **-------** |

**Encl: snapshots of the daily class activities**

|  |  |
| --- | --- |
| **Coding Challenges** | |
| **Problem Statement: Prog1(java)** | |
| **Status: Completed** | |
| **Uploaded the report both in GitHub & Slack** | Yes |

**Encl: snapshots of your response to challenge.**

1. Write a program to find given two trees are mirror or not.

class Node

{

int data;

Node left, right;

public Node(int data)

{

this.data = data;

left = right = null;

}

}

public class BinaryTree

{

Node a, b;

boolean areMirror(Node a, Node b)

{

if (a == null && b == null)

return true;

if (a == null || b == null)

return false;

return a.data == b.data

&& areMirror(a.left, b.right)

&& areMirror(a.right, b.left);

}

public static void main(String[] args)

{

BinaryTree tree = new BinaryTree();

Node a = new Node(1);

Node b = new Node(1);

a.left = new Node(2);

a.right = new Node(3);

a.left.left = new Node(4);

a.left.right = new Node(5);

b.left = new Node(3);

b.right = new Node(2);

b.right.left = new Node(5);

b.right.right = new Node(4);

if (tree.areMirror(a, b) == true)

System.out.println("Yes");

else

System.out.println("No");

}

}

Output:

