

# Daily Coding Practice-2

## 1. Palindrome Linked List:

```
class Solution {
    boolean isPalindrome(Node head) {
        if (head == null || head.next == null) return true;
        Node slow = head;
        Node fast = head;

        while (fast != null && fast.next != null) {
            slow = slow.next;
            fast = fast.next.next;
        }
        Node secondHalf = reverseList(slow);
        Node firstHalf = head;
        while (secondHalf != null) {
            if (firstHalf.data != secondHalf.data) {
                return false;
            }
            firstHalf = firstHalf.next;
            secondHalf = secondHalf.next;
        }

        return true;
    }
    private Node reverseList(Node head) {
        Node prev = null;
        while (head != null) {
            Node nextNode = head.next;
            head.next = prev;
            prev = head;
            head = nextNode;
        }
        return prev;
    }
}
```

## Output:

### Output Window

[Compilation Results](#) [Custom Input](#) [Y.O.G.I. \(AI Bot\)](#)

**Problem Solved Successfully** ✓[Suggest Feedback](#)

Test Cases Passed  
**1112 / 1112**

Attempts : Correct / Total  
**1 / 1**  
Accuracy : 100%

Points Scored   
**4 / 4**  
Your Total Score: 10

Time Taken  
**1.94**

**Solve Next**

[Intersection Point in Y Shaped Linked Lists](#) [Flattening a Linked List](#)

## 2.Balanced Tree Check:

Courses

Tutorials

Jobs

Practice

Contests

Problem

Editorial

Submissions

Comments

Output Window

[Compilation Results](#) [Custom Input](#) [Y.O.G.I. \(AI Bot\)](#)

**Problem Solved Successfully** ✓[Suggest Feedback](#)

Test Cases Passed  
**1120 / 1120**

Attempts : Correct / Total  
**1 / 1**  
Accuracy : 100%

Points Scored   
**2 / 2**  
Your Total Score: 12

Time Taken  
**0.42**

**Solve Next**

[Height of Binary Tree](#) [Minimum Depth of a Binary Tree](#) [Array to BST](#)

Java (1.8) Average Time: 20m [Start Timer](#)

```
125 Node(int d)
126 {
127     data = d;
128     left = right = null;
129 }
130 */
131
132 class Tree {
133     private int checkHeight(Node root) {
134         if (root == null) {
135             return 0;
136         }
137         int leftHeight = checkHeight(root.left);
138         int rightHeight = checkHeight(root.right);
139         if (leftHeight == -1 || rightHeight == -1) {
140             return -1;
141         }
142         if (Math.abs(leftHeight - rightHeight) > 1) {
143             return -1;
144         }
145         return 1 + Math.max(leftHeight, rightHeight);
146     }
147     boolean isBalanced(Node root) {
148         return checkHeight(root) != -1;
149     }
150 }
151
152
153
```

### 3.Triplet Sum In Array:

Output Window

Compilation Results

Custom Input

Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓

Test Cases Passed

125 / 125

Attempts : Correct / Total

1 / 1

Accuracy : 100%

Points Scored ⓘ

4 / 4

Your Total Score: 16 ↑

Time Taken

0.17

Solve Next

Java (1.8)

Average Time: 15m

Start Timer

```
1 // } Driver Code Ends
2
3
4 // User function Template for Java
5
6 class Solution {
7     public static boolean find3Numbers(int arr[], int n, int x) {
8         Arrays.sort(arr);
9         for (int i = 0; i < n - 2; i++) {
10             int target = x - arr[i];
11             int left = i + 1;
12             int right = n - 1;
13             while (left < right) {
14                 int currentSum = arr[left] + arr[right];
15                 if (currentSum == target) {
16                     return true;
17                 } else if (currentSum < target) {
18                     left++;
19                 } else {
20                     right--;
21                 }
22             }
23         }
24         return false;
25     }
26 }
```