

# Read Me

## Contents

<b>Question 01</b> .....	2
<b>Question 02</b> .....	2
<b>Question 03</b> .....	2
<b>Question 04</b> .....	2
<b>Question 05</b> .....	2
<b>Question 06</b> .....	3

### Question 01

The array given was sorted so take it from center and started inserting the nodes that help to minimize the tree height. First mid value was taken then then the detrimental loop was done to insert lower than mid and incremental to insert greater than mid.

### Question 02

In question the all nodes were extracted by the breath first search. The tree was passed and linked list was made level wise and was printed and the height was returned. The list was equal to the depth of tree.

### Question 03

The height of each node was seen and then was calculated if it was equal to or less than 1 than is balance thus recursively check that whether the nodes are balance or not if not then return.

### Question 04

The nodes were extracted and were stored in a array than on each level swapping was done. Such that swapping was done between the siblings. The same level siblings with same parent were swap and possible BST combinations was made.

### Question 05

First the node was found in the tree if found then started to see from that the subtree otherwise return false as not possible subtree. Then data was matched and height recursively.

## Question 06

The nodes were extracted and paths were figured out from each node if any encountered than the count was incremented and was done recursively.