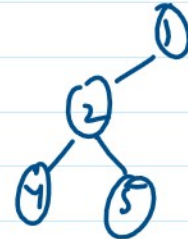


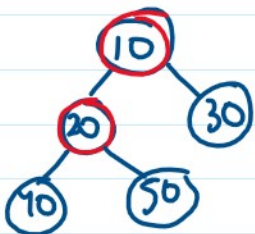
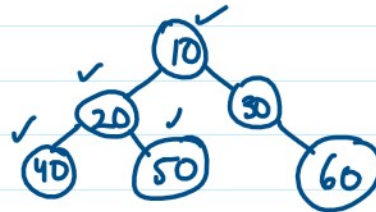
$0 \rightarrow \text{depth}$
 $\text{depth} \times n$



10, 20, 30, 40, 50, 60, 70

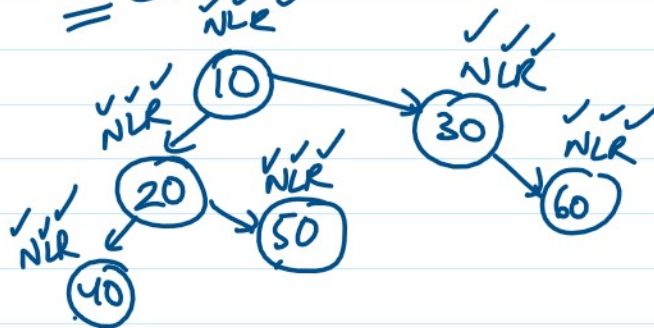
$\{10^0, 20^1, 30^2, 40^3, 50^4, \overset{\text{null}}{\underset{\text{idx}}{-1}}, 60^6\}$ L.O.T

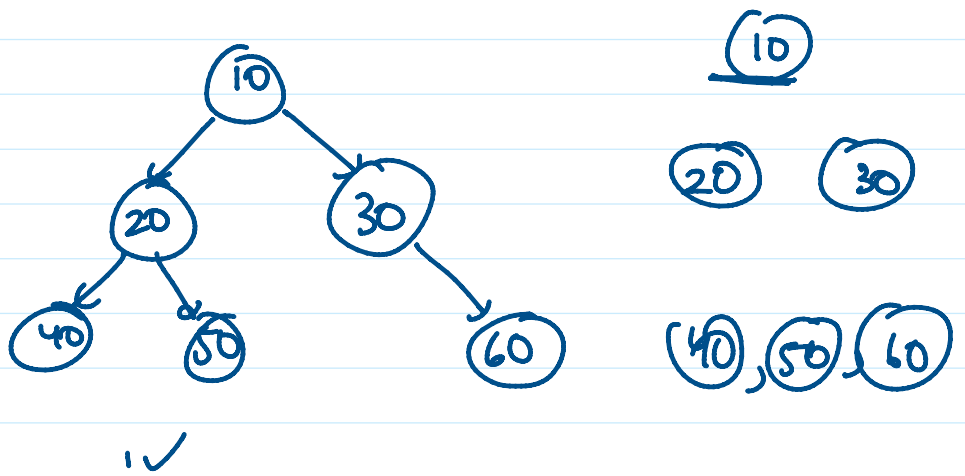
root = 10



$(10, (20, (40, -1, -1, 50, -1, -1), (30, -1, (60, -1, -1)))$

NLR





Node GT Node

Node left
Node right

```

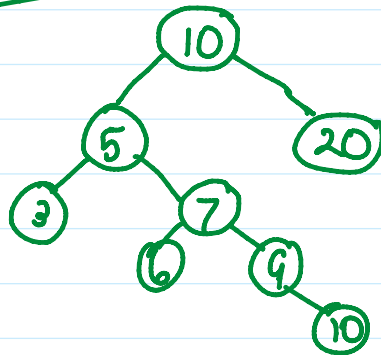
int data;
ArrayList<Node> children;
}
  
```

B.S.T
→ Binary Search Tree

B.T + $\left(\begin{array}{c} \text{Left} \\ \text{Subtree} \\ \text{Data} \end{array} \right) \leq \text{node.data} \leq \left(\begin{array}{c} \text{Right} \\ \text{Subtree} \\ \text{Data} \end{array} \right)$



LNR



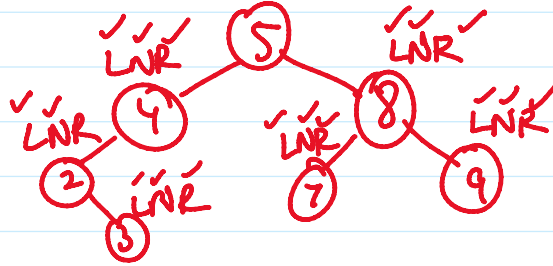
10, 5, 3, 7, 6, 9, 10, 20

3, 6, 10, 9, 7, 5, 20, 10

3, 5, 6, 7, 9, 10, 10, 20

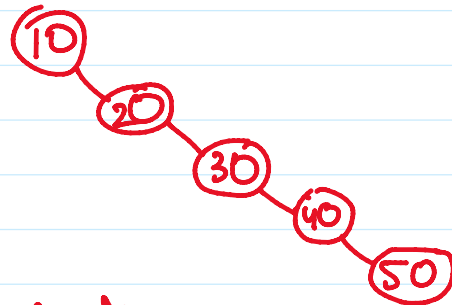
* In BST ⇒ Inorder is always sorted

✓ In BST - inorder is always sorted

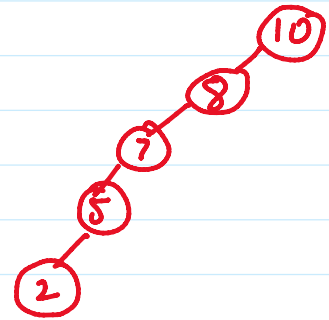


2, 3, 4, 5, 7, 8, 9,

1

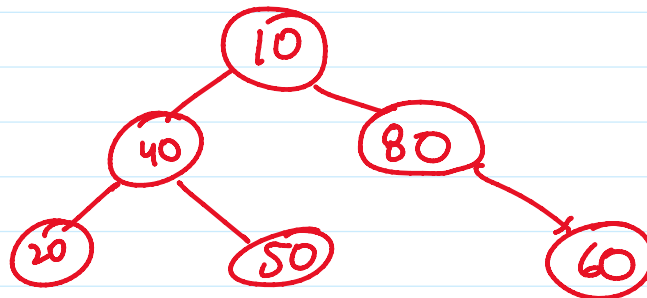
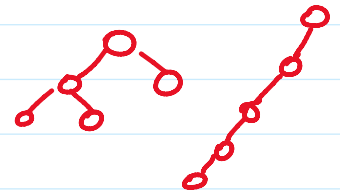


Right Skewed Tree



Left Skewed Tree

$O(n)$?? $O(h)$



1) \Rightarrow Inorder ✓

2) Property

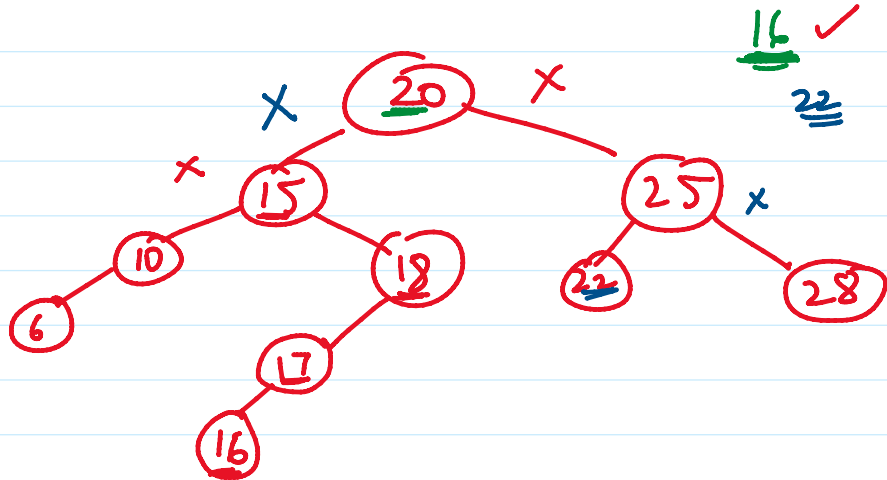
L.C ✓

R.C ✓

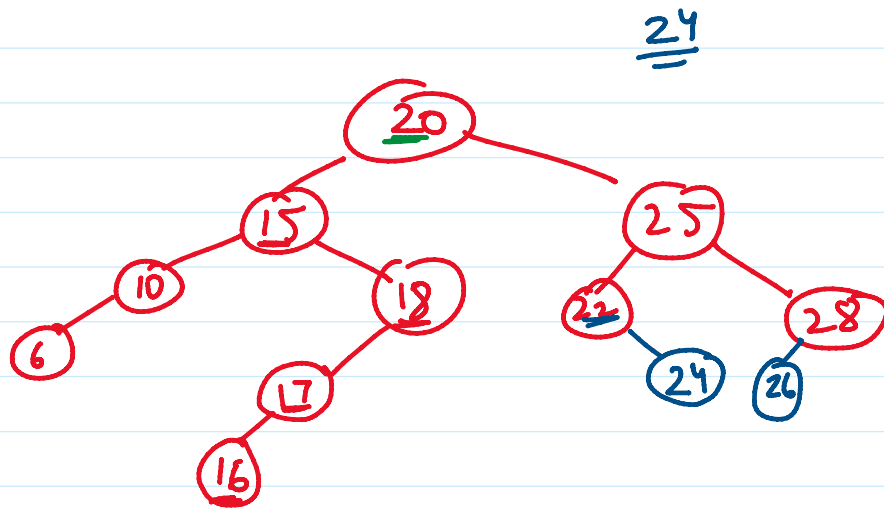
L.st. Max \leq Node \leq R.st Min

Size ✓
 HT ✓
 Diameter ✓
 is Balanced ✓

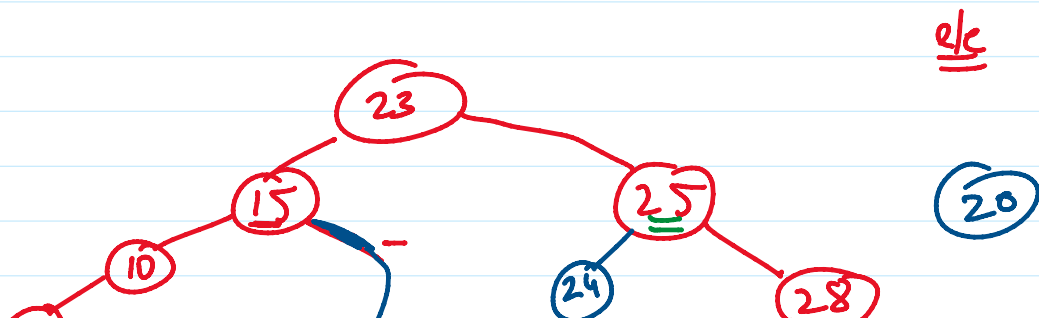
Find
Add
 Max
 Min

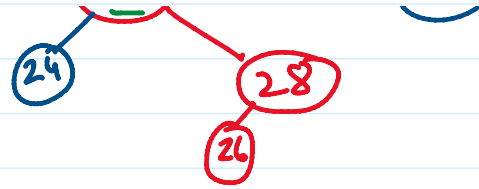
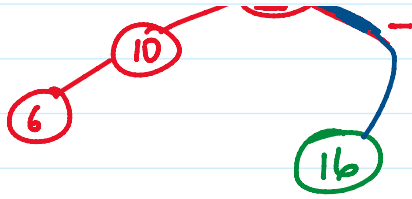


Find
 $O(n) \rightarrow O(h)$



6, 10, 15, 16, 18, 20, 23, 24, 25, 26, 28





→ find — ^x null BSI
 ↙
 1) Leaf node
 2) 1 child
 3) 2 children

$[-10, -5, 0, 5, 9]$

1 +3 +4+4

