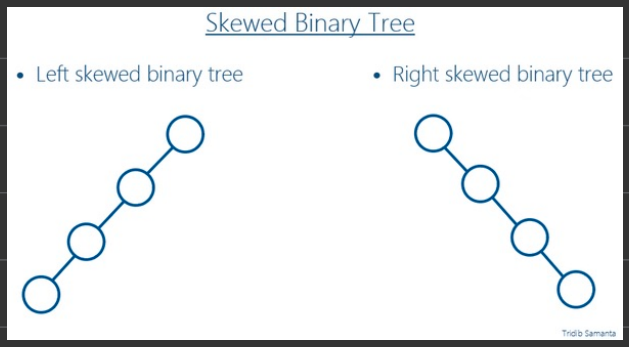
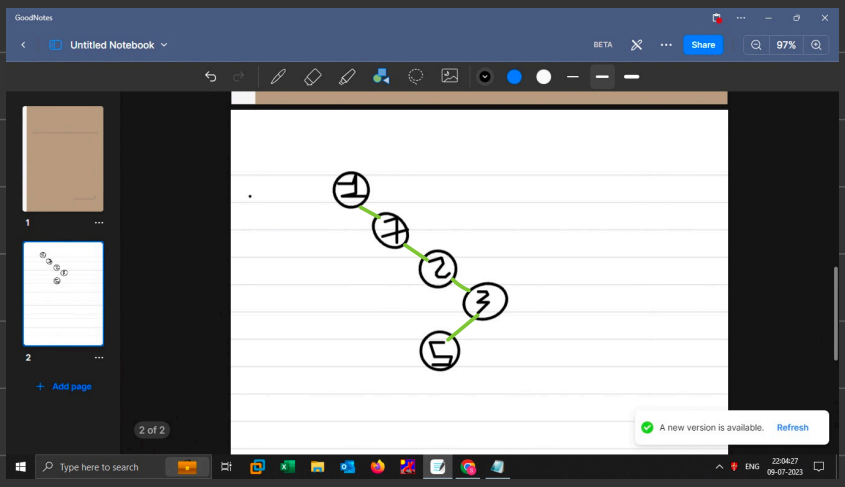
9 July 2023

Skewed Tree: The binary tree in which each node has either one or no child



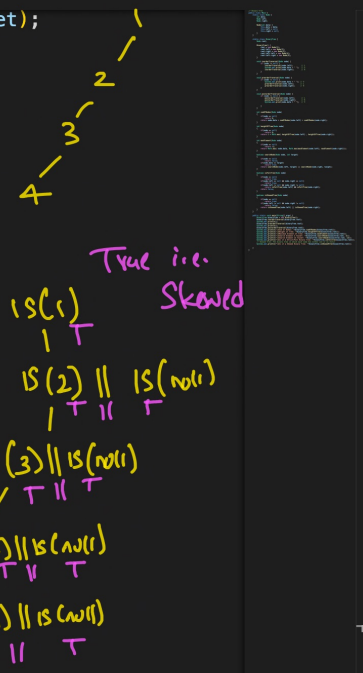


NS 9 July 2023

Main.java

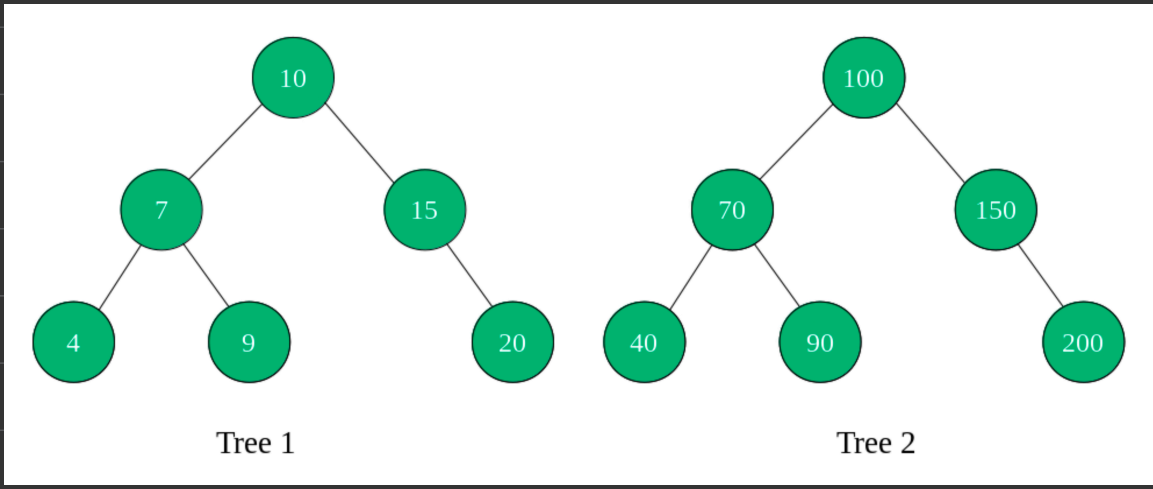
Main.java > Main > main(String[])

```
78              return searchNode(node.left, target) || searchNode(node.right, target);
79          }
80
81          boolean isFullTree(Node node)
82          {
83              if(node == null)
84                  return true;
85              if(node.left == null && node.right == null)
86                  return true;
87              if(node.left != null && node.right != null)
88                  return isFullTree(node.left) && isFullTree(node.right);
89              return false;
90          }
91
92          boolean isSkewedTree(Node node)
93          {
94              if(node == null)
95                  return true;
96              if(node.left != null && node.right != null)
97                  return false;
98              return isSkewedTree(node.left) || isSkewedTree(node.right);
99          }
100     }
101
        Run | Debug
102     public static void main(String[] args) {
103         BinaryTree binaryTree = new BinaryTree();
```



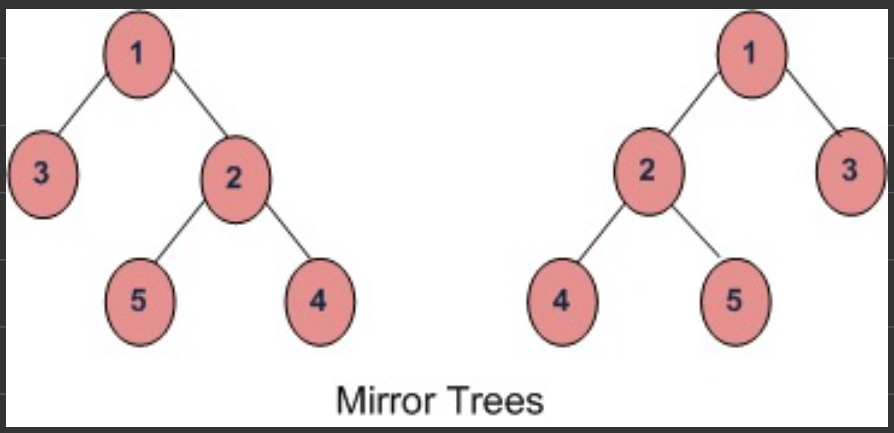
0 0 Live Share Ln 116, Col 84 Spaces: 4 UTF-8 LF Java

Similar/Identical Trees: Two binary trees are called similar if both are having a similar structure but the elements in both the trees are different.

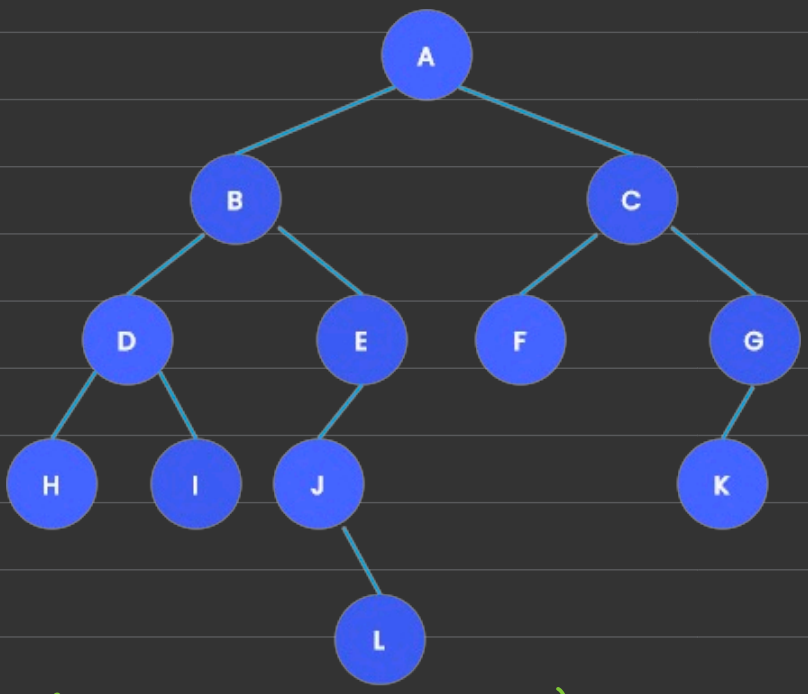



```
boolean isSimilar(Node root1, Node root2)
{
    if(root1 == null && root2 == null)
        return true;
    if(root1 == null || root2 == null)
        return false;
    return isSimilar(root1.left, root2.left) && isSimilar(root1.right, root2.right);
}
```

https://www.geeksforgeeks.org/check-if-two-trees-are-mirror/



Sr. No.	Key	BFS	DFS
1	Definition	BFS, stands for Breadth First Search.	DFS, stands for Depth First Search.
2	Data structure	BFS uses Queue to find the shortest path.	DFS uses Stack to find the shortest path.
3	Source	BFS is better when target is closer to Source.	DFS is better when target is far from source.
4	Suitablity for decision tree	As BFS considers all neighbour so it is not suitable for decision tree used in puzzle games.	DFS is more suitable for decision tree. As with one decision, we need to traverse further to augment the decision. If we reach the conclusion, we won.
5	Speed	BFS is slower than DFS.	DFS is faster than BFS.
6	Time Complexity	Time Complexity of BFS = O(V+E) where V is vertices and E is edges.	Time Complexity of DFS is also O(V+E) where V is vertices and E is edges.



BFS (Left to Right, Top to Bottom): A, B, C, D, E, F, G, H, I, J, K, L

In-order Traversal - H, D, I, B, J, L, E, A, F, C, K, G

DFS / Pre-order Traversal - A, B, D, H, I, E, J, L, C, F, G, K

Post-order Traversal - H, I, D, L, J, E, B, F, K, G, C, A