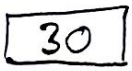
2-3 Tree
(OUTPLAY)

Bryon Tamin

Case 1 2-3 Tree. Max child = 3
Max keys = 2
Min keys = 1

1. Insert 30

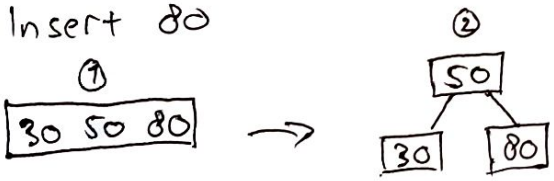


1. Start searching for node from the root node.
2. Because there are no keys in root node, we can insert 30 to root node.

2. Insert 50

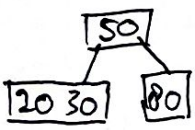
1. Start searching for node from the root node.
2. Insert 50 in root node. in sorted form.

3. Insert 80



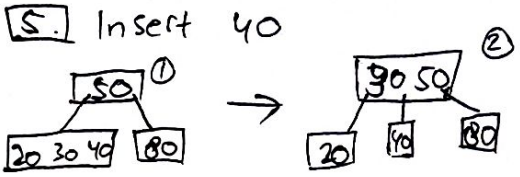
1. Insert 80 in sorted form.
2. Because max ~~child~~ keys in one node is 2, but we have 3. we split it and take 50 as the median.

4. Insert 20



1. Search for leaf node, because 20 < 50 it goes left. left node is a leaf so insert 20 in it. Insert in sorted form.

5. Insert 40



1. Search for leaf node, 40 < 50 so it goes left, now it is a leaf node so insert 40 in sorted form.
2. The node exceed max keys so it will split and take 30 as median, 30 then merge with root node in sorted form.

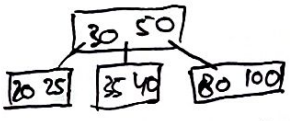
6. Insert 100

1. Search for leaf node, 100 > 50 so it goes right, now it is in leaf node so insert 100 in sorted form.

7. Insert 25

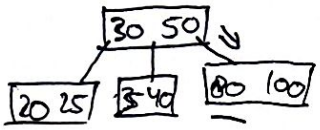
1. Search for leaf node, 25 < 30 so it goes to the left, now it is in leaf node so insert 25 in sorted form.

8. Insert 35



1. Search for leaf node, 35 > 30 but 35 < 50 so it goes between them. Now it is in leaf node so insert 35 in sorted form.

9. Search 80

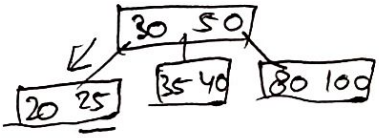


1. 80 > 30, 80 > 50 so it goes to the right child, 80 == 80
2. found 80.

10. Search 100

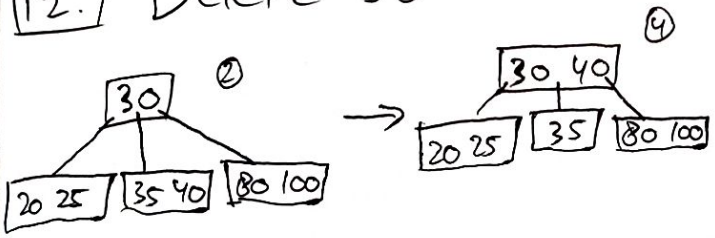
1. 100 > 30, 100 > 50, traverse to right child.
2. 100 > 80, 100 == 100 found 100.

11. Search 25.



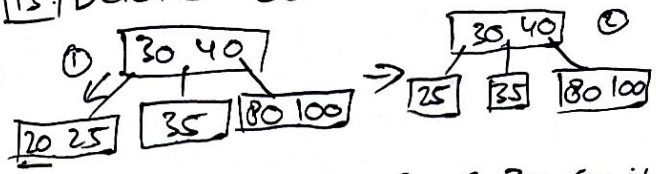
1. 25 < 30, traverse to left child.
2. 25 > 20, 25 == 25 found 25.

12. Delete 50



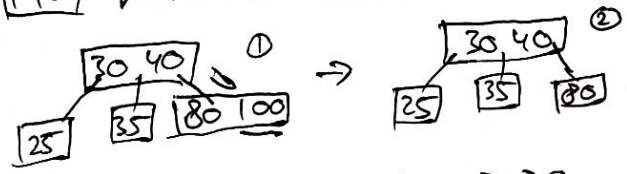
1. Search for 50, 50 > 30, 50 == 50 found 50.
2. Delete 50.
3. root node has 3 child but only has 1 key. Search for 50 predecessor.
4. 30 < predecessor < 50 so it goes to the node below it. take the rightmost value which is 40 then take 40 as 50 replacement in root node.

13. Delete 20



1. Search for 20, 20 < 30 so it goes to left child. 20 == 20, found 20.
2. Delete 20.

14. Delete 100



1. Search for 100, 100 > 30, 100 > 40, it goes to right child node. 100 > 80, 100 == 100 found 100.
2. Delete 100.