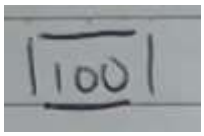
Insert: 100, 20, 60, 50, 30

Insert

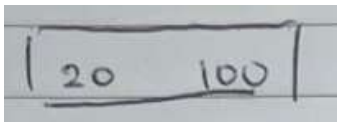
B-TREE

- 100



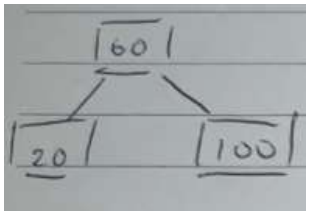
First, we check if the tree is empty, the tree is empty so, 100 will be inserted as the root.

- 20



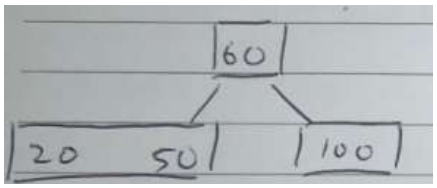
Check if we have already passed the limit of keys, it hasn't so, insert 20 in the left side of 20, because the node is sorted.

- 60



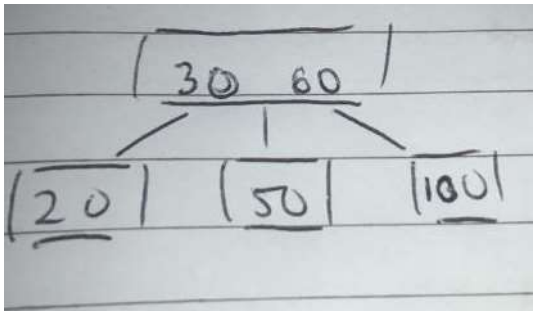
The node has reach it's limit so we used the median here(60), to split and by doing so, we keep the properties of a B-Tree.

- 50



First, check if 50 > / < 60, 50 is smaller so we traverse left, then check if we have already passed the limit of keys, it hasn't so, insert 50 in the left of 20.

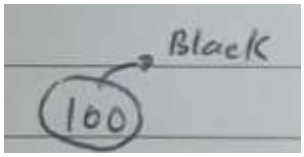
- 30



First , we check if 30 > / < 60, it's smaller, it's smaller so traverse left, then check if the left children node(20 and 50) is full, it's full so return 30 back and split the leafs.

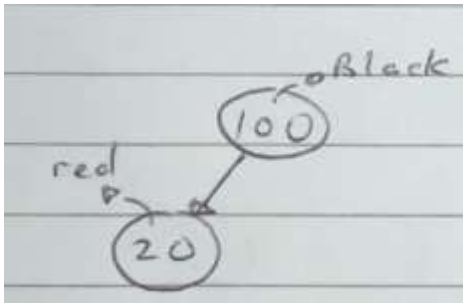
Red-Black Tree

- 100



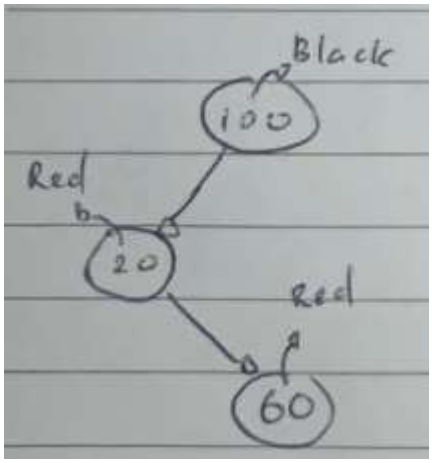
1. Newly inserted nodes will be **RED**
2. If x is the root, change the colour of x as **BLACK**

- 20

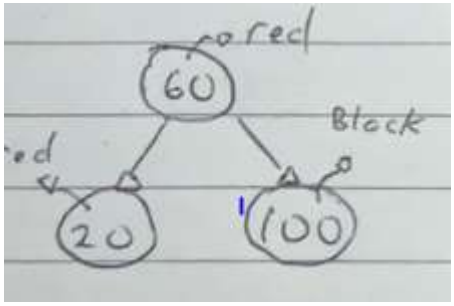


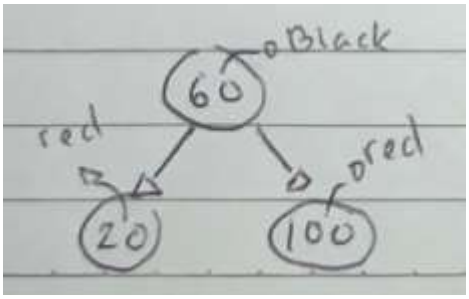
1. Check if 20 is > / < 100, it's smaller than 100 so traverse left.
2. Newly inserted nodes will be **RED**

- 60



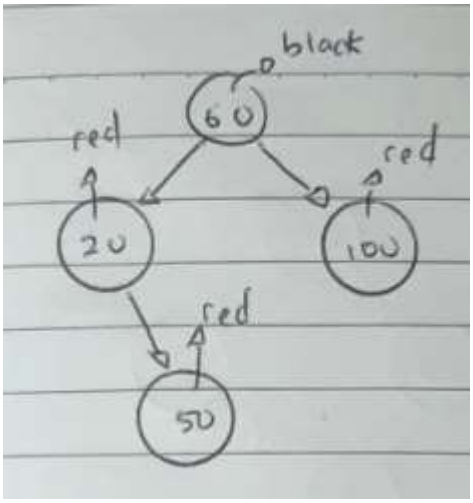
1. Check if 20 is > / < 100, it's smaller than 100 so traverse left.
2. Newly inserted nodes will be **RED**
3. The last two nodes are red, so we must do a left-right rotation, the next pictures is the step and the end result of the rotation.



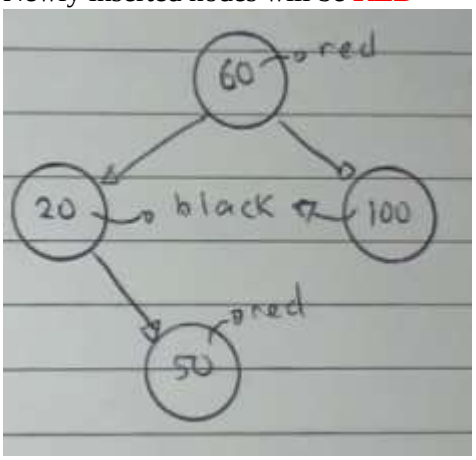


3. Newly inserted nodes will be **RED,** so we swapped the colors

- 50

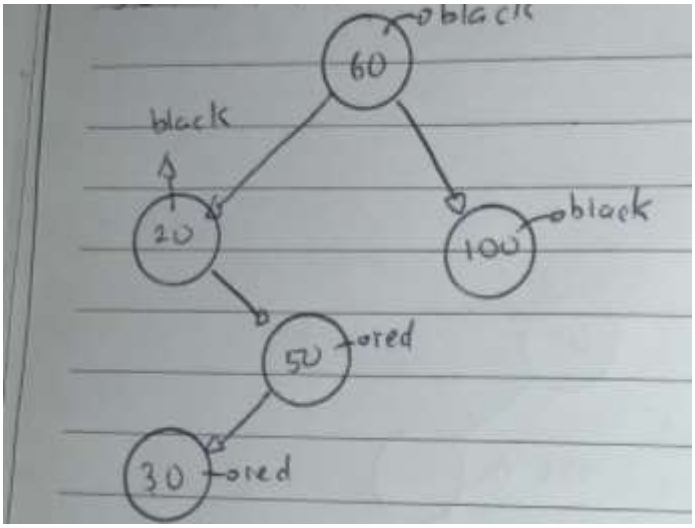


1. First, we need to check is 50 > / < 60, 50 is smaller so we traverse left
2. then check is 50 > / < 20, 50 is bigger so traverse right.
3. Newly inserted nodes will be **RED**

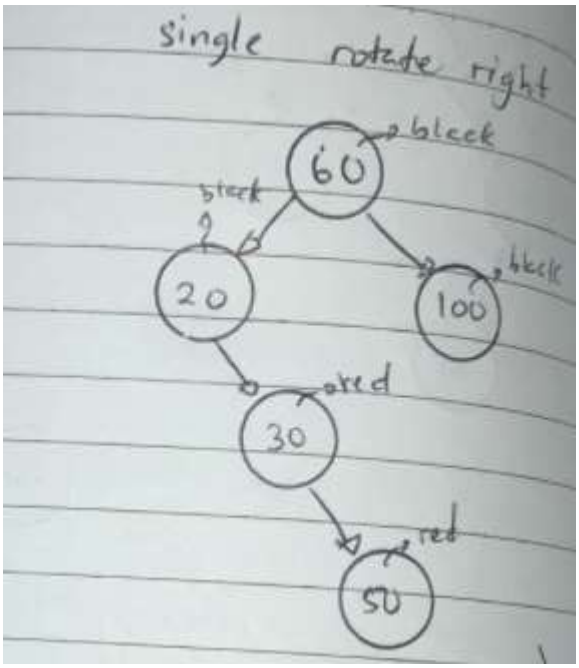


4. After making the new inserted nodes red, there's a problem, both parent and child are red, so we keep 20's parent color and swapped the color, keeping both the conditions(parent and child can't be red and the black nodes each ways must be the same).

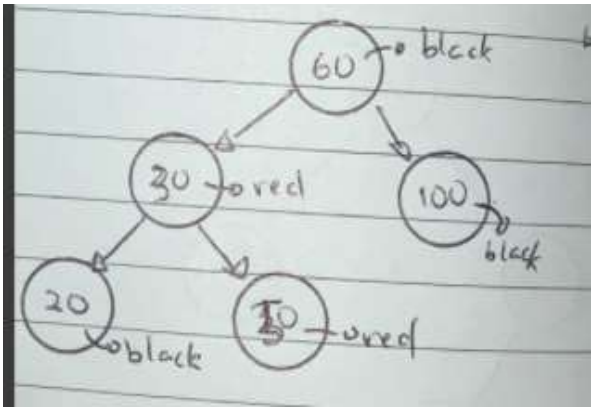
- 30



1. First, we need to check is 30 > / < 60, 30 is smaller so we traverse left
2. then check is 30 > / < 20, 50 is bigger so traverse right.
3. Newly inserted nodes will be **RED**



5. After making the new inserted nodes red, there's a problem, both parent and child are red, so first we need to do a left rotation. the next picture shows the end result after doing a left rotation.



6. After being done with the rotation, there's another problem, the leaf 20 is black making the black nodes we need to traverse to each leaf is unequal, so we need to swap colors.