

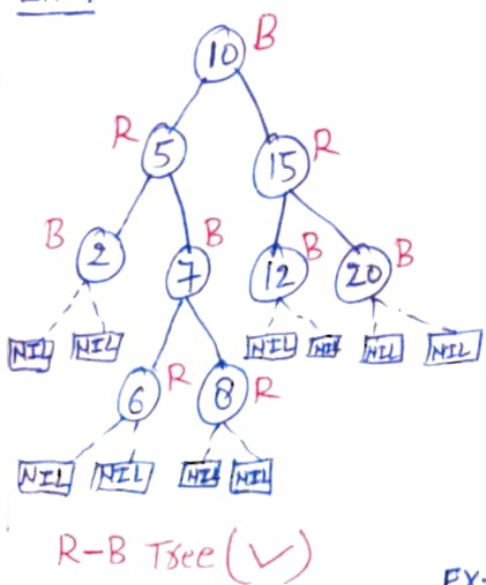
## Red-Black Trees

A red-black tree is a binary search tree with one extra bit of storage per node: its color, which can be either RED or BLACK. By constraining the node colors on any simple path from the root to a leaf, red-black trees ensures that no such path is more than twice as long as any other, so that the tree is approximately balanced.

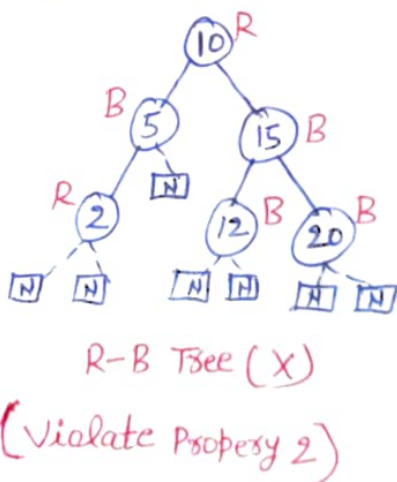
A red-black tree is a binary tree that satisfies the following red-black properties:  $\rightarrow$  Red-Black tree is a self-balancing BST.

1. Every node is either red or black.
2. The root is black.
3. Every leaf (NIL) is black.
4. If a node is red, then both its children are black.
5. For each node, all simple paths from the node to descendant leaves contain the same number of black nodes.

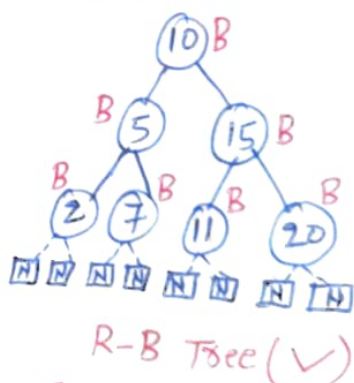
### EX-1



### EX-2

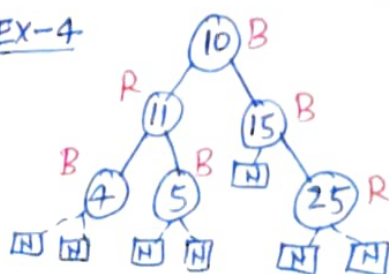


### EX-3



Even not a single red color node in a tree.

### EX-4



Even though it follows all the properties of R-B tree but it's not a BST.

# Insertion in Red-Black Tree

## Algorithm:

- ① If tree is empty, ~~create~~ create new node as root node with color Black.
- ② If tree is not empty, create new node as leaf node with color Red.
- ③ If Parent of new node is 'black' then exit.
- ④ If Parent of new node is 'red' then check the color of parent's sibling of new node:  
or (uncle)
  - Ⓐ If color is black or null then do suitable rotation & recolor.
  - Ⓑ If color is red then recolor both parent and sibling & also check if parent's parent of new node is not root node then recolor it & recheck.

Array: 10, 18, 7, 15, 16, 30, 25, 40, 60, 20, 1, 70

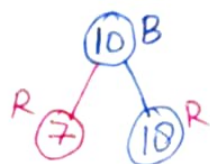
Step 1: Insert element 10



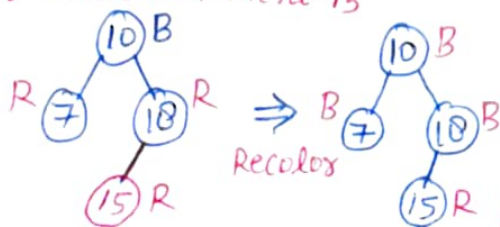
Step 2: Insert element 18



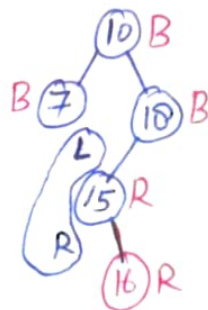
Step 3: Insert element 7



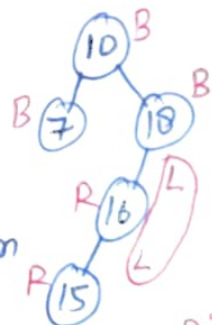
Step 4: Insert element 15



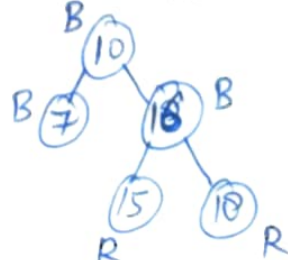
Step 5: Insert element 16



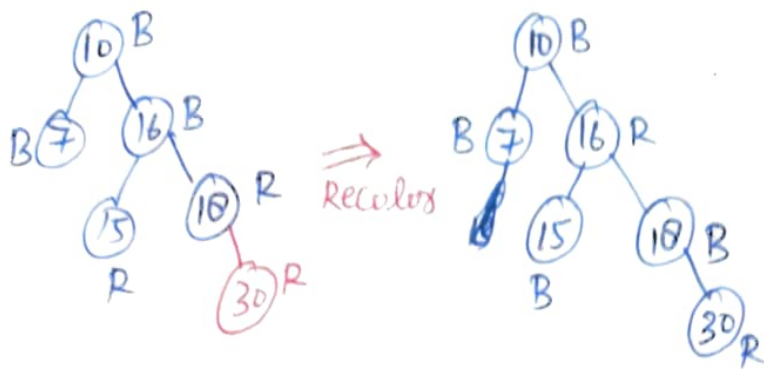
⇒ L-R Rotation



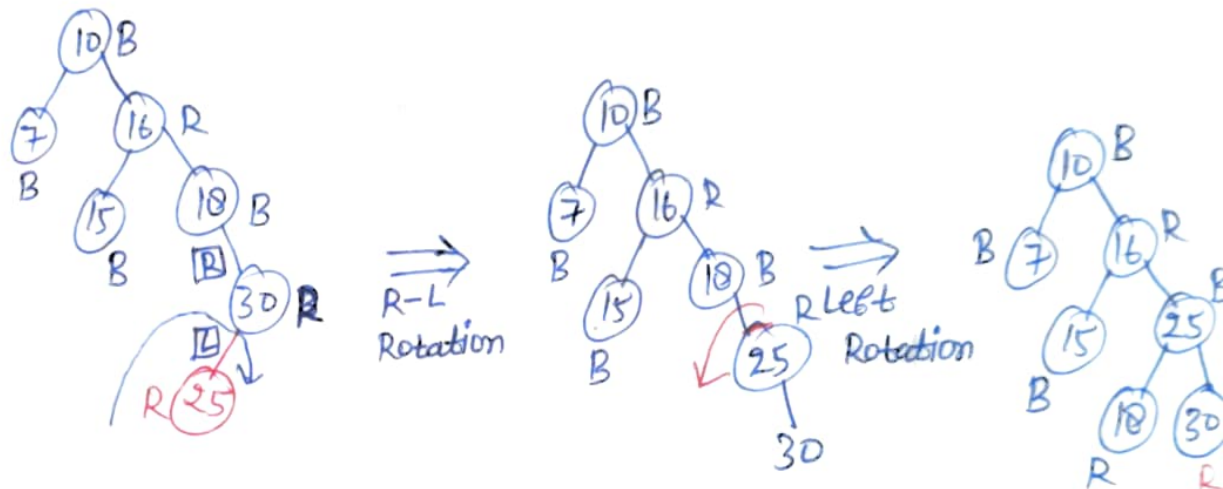
Right Rotation



Step 6: Insert element 30



Step 7: Insert element 25



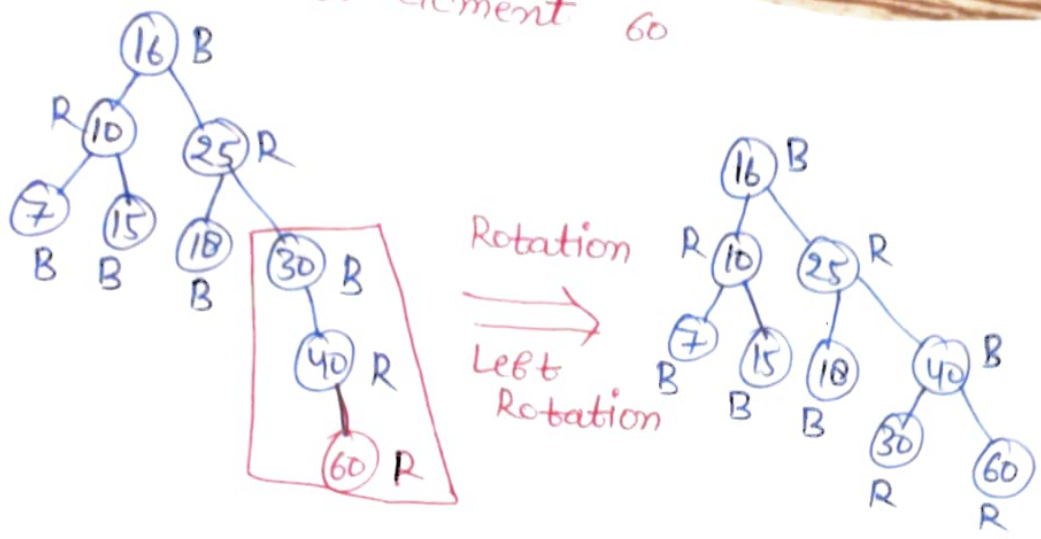
Step 8: Insert element 40



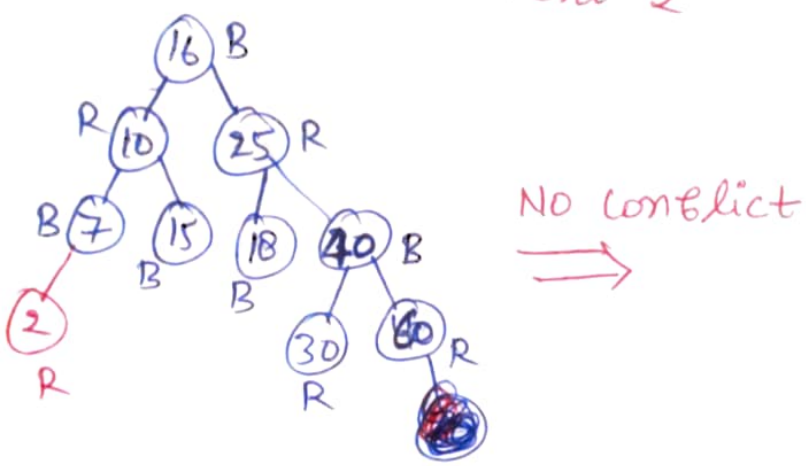
~~Step 9: Insert element~~



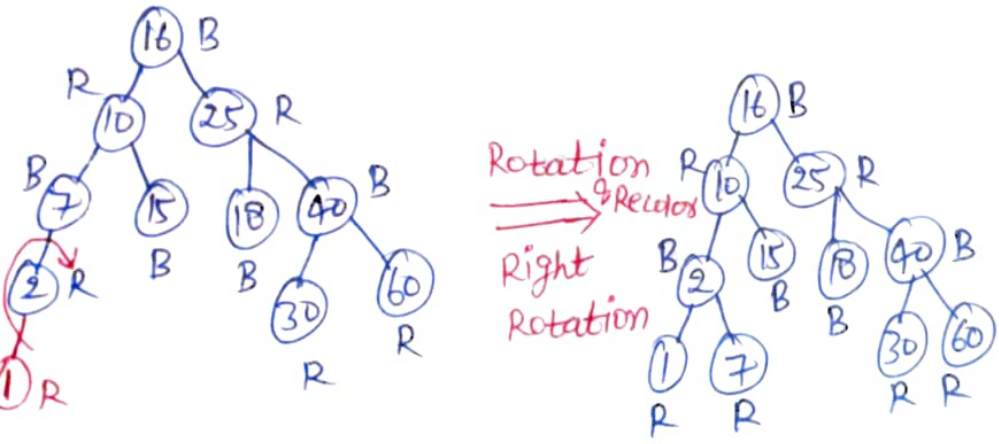
Insert element 60



Step 10: Insert element 2



Step 11: Insert element 1



Step 12: Insert element 70

