Deleting a node from a RB tree

Point of
- RB trees is to maintain height balance of subtrees.
- Important: understanding of tree manipulations

BST Delete

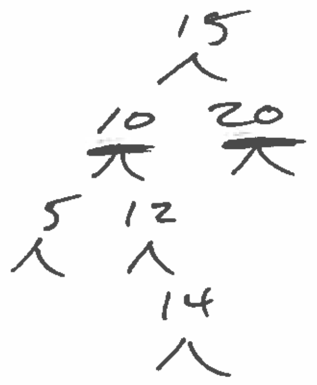
- No children - delete node
- One child - replace node w/ child
- Two children - replace w/ min in right subtree

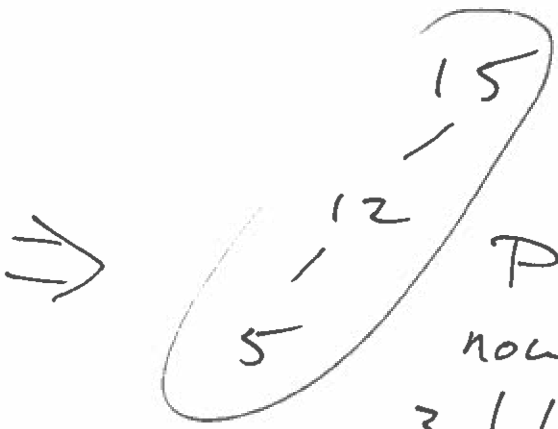
Violation

Violation occurs if node's replacement is black.
can change the # of black nodes on a path

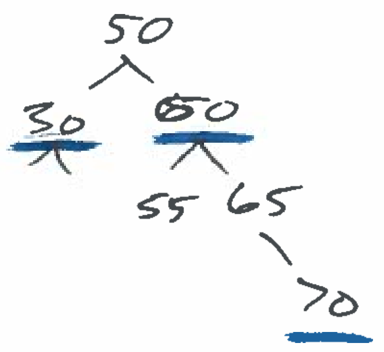
Ex:

Delete 10.
Replaced by 12
12 replaced by 14





Path now has 3 black nodes instead of 2.



Delete 60

nodeColor = RED

2 children

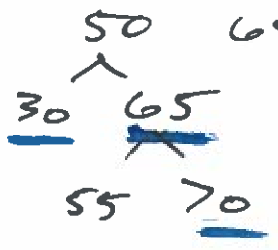
Min = 65

nodeColor = BLACK

X = 70

X. parent = 65

min. color = node.color

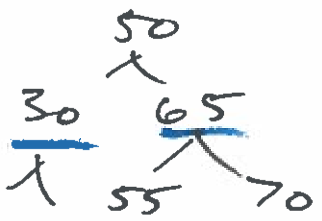


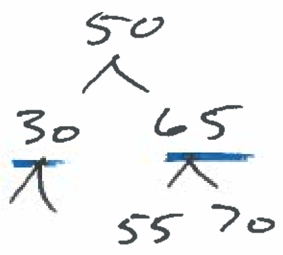
65 now RED

rbBalance(70)

X is RED

recolor X to BLACK





Delete the 55

what is x?

What is nodeColor?

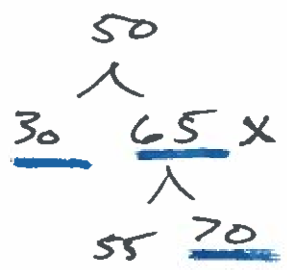
is rbBalance called?

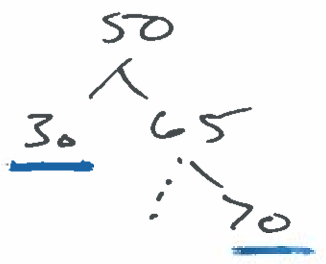
x = 55

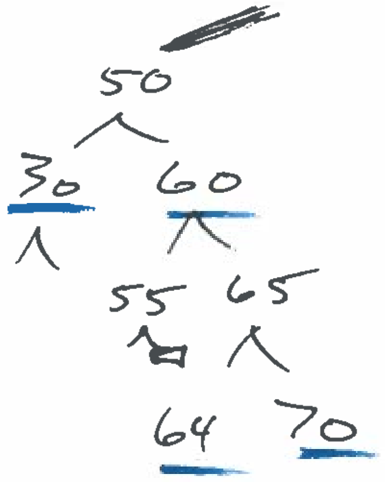
nodeColor = BLACK

rbBalance(55)

s = 70







Delete 50

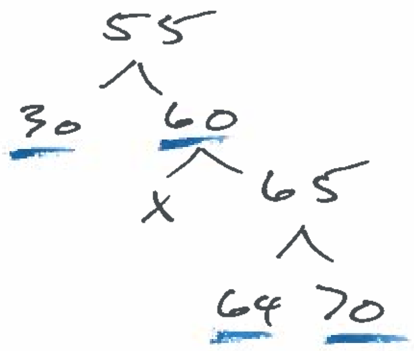
nodeColor = BLACK

Min = 55

nodeColor = BLACK

X = nullNode

Min.color = BLACK



rbBalance(x) X = nullNode

S = 65, BLACK

[illegible]

Start here on Wednesday

```
RBDelete(value)
  node = search(value)
  nodeColor = node.color        //nodeColor is color of node to
                                          delete
  if(node != root)
     if (no children)
        x = node.parent.left   //assumes node is left
                                                    child
     elseif(two children)
        min = treeMin(node.right)
        nodeColor = min.color      //nodeColor is color
                                         of replacement
        x = min.rightChild  }  x could be nullNode.
        x.parent = min      }
        // code to replace node with min from
                 BST delete
        min.color = node.color   //change color of
                                        replacement
     else(one child)
        x = node.left                 }
        node.parent.left = x          }   x is node's
        x.parent = node.parent        }   replacement
  else //handle root
  if nodeColor == BLACK
       rbBalance(x)                  x can be nullnode,
                                     min's right child, or
  delete node                        node's replacement
```