ARAVINDH R
1BM18CS145

ADS Lab 8

```
func insert(key):
        node = new Node(key)
        node.item = key
        node.left = node.right = NULL
        node.color = 1

        y = None
        x = root

        while x != NULL:
              y = x
              if node.item < x.item:
                     x = x.left
              else:
                     x = x.right
        node.parent = y
        if y == NULL:
              root = node
        else if node.item < y.item: y.left = node
        else : y.right = node

      if node.parent == NULL:
            node.color = 0
            return

      fix_insert( node)
```

```
func fix-insert(k):
    while k.parent.color == 1:
        if k.parent == k.parent.parent.right:
            u = k.parent.parent.left
            if u.color ==1:
                u.color = 0
                k.parent.color = 0; k.parent.parent.code = 1
                k = k.parent.parent

        else:
            if k == k.parent.left : k = k.parent
            self.right-rotate(k)
            k.parent.color = 0; k.parent.parent.color = 1
            left-rotate(k.parent.parent)
       else: u = k.parent.parent.right

       if u.color == 1:
            u.color = 0
          k.parent.color = 0; k.parent.parent.color = 1
          right-rotate (k.parent.parent)

      if k == root:
         break

func left-rotate(x):
   y = x.right; ~~y.~~ x.right = y.left
   if y.left != NULL : y.left.parent = x
   if not x.parent : root = y
  if x == x.parent.left : x.parent = y
   y.left = x; x.parent = y
```

similar to right rotate