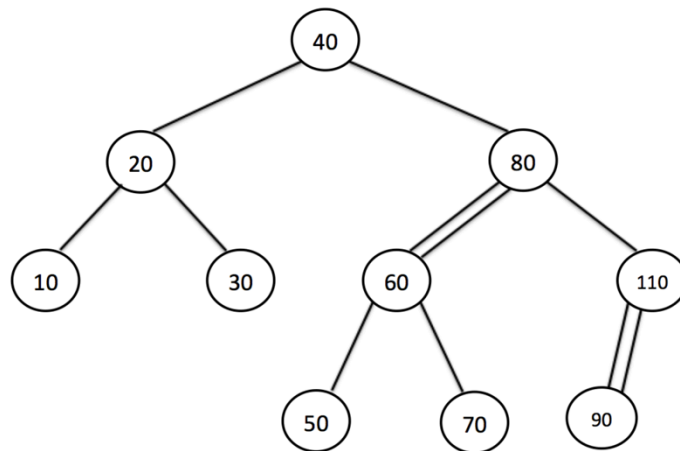


**CMPE 242**  
**Spring 2021**  
**Hands-On Activity 11**

1. Draw a 12-node AVL tree of **maximum** height. (Hint: consider the balance property of the AVL tree).
  
  
  
  
  
  
  
  
  
  
2. Insert 100 to the following Left-Leaning Red-Black Tree using the algorithm that we mentioned in class. Show the intermediate steps of this insertion and draw the resulting Red-Black Tree. (Note: Red links are shown as double lines in the tree).



**Assume that we have the following API for the BST, for Questions 1-2 below.**

```
public class BinarySearchTree {  
  
    private Node root;  
  
    private class Node {  
        private int key;           % key  
        private Node left;         % left child  
        private Node right;        % right child  
  
        ...  
        private _____ rotateLeft( Node n);  
        public boolean isAVL();  
    }  
}
```

3. Write a private Java method that performs single left rotation on a given node n.

```
private _____ rotateLeft( Node n)
```

4. Write a public method that checks whether the given binary search tree is an AVL tree.

```
public boolean isAVL( )
```