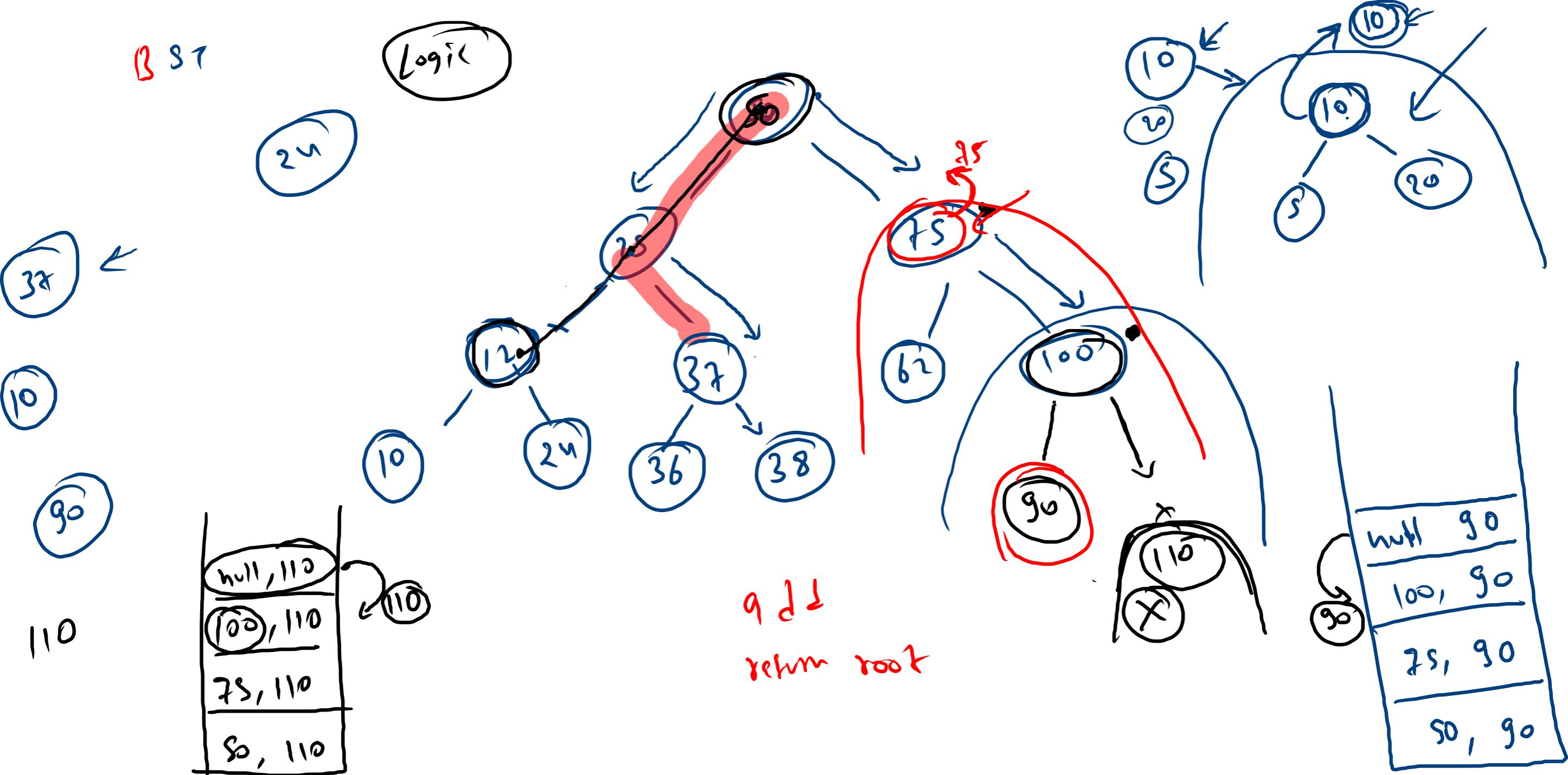
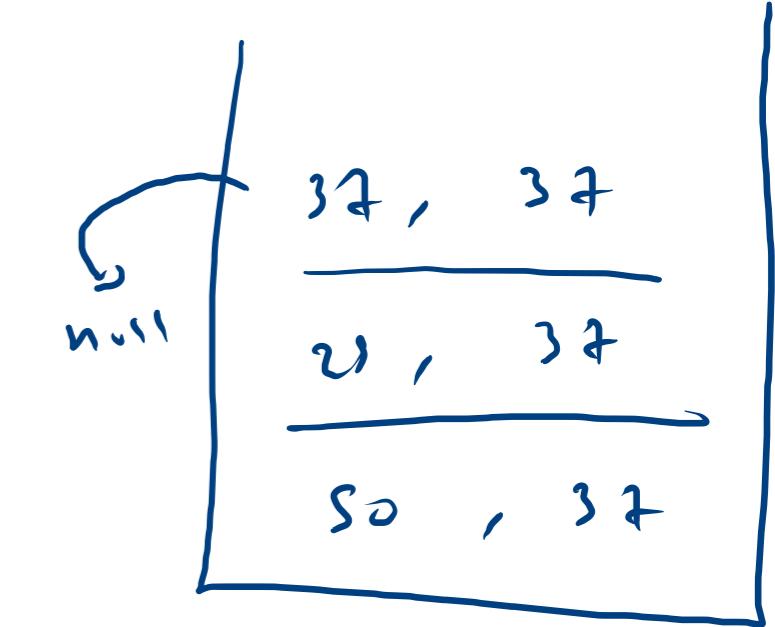
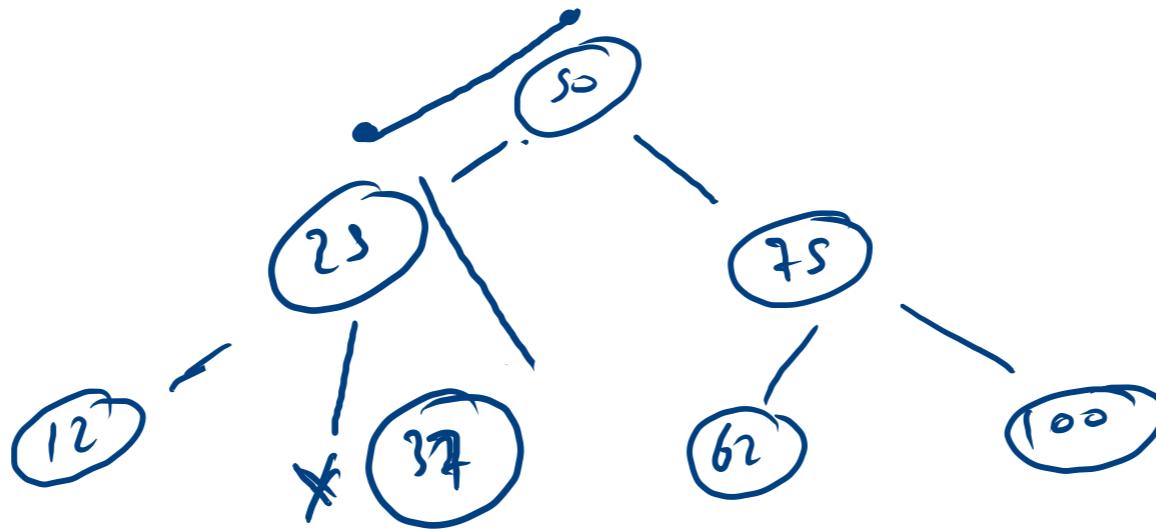


BST

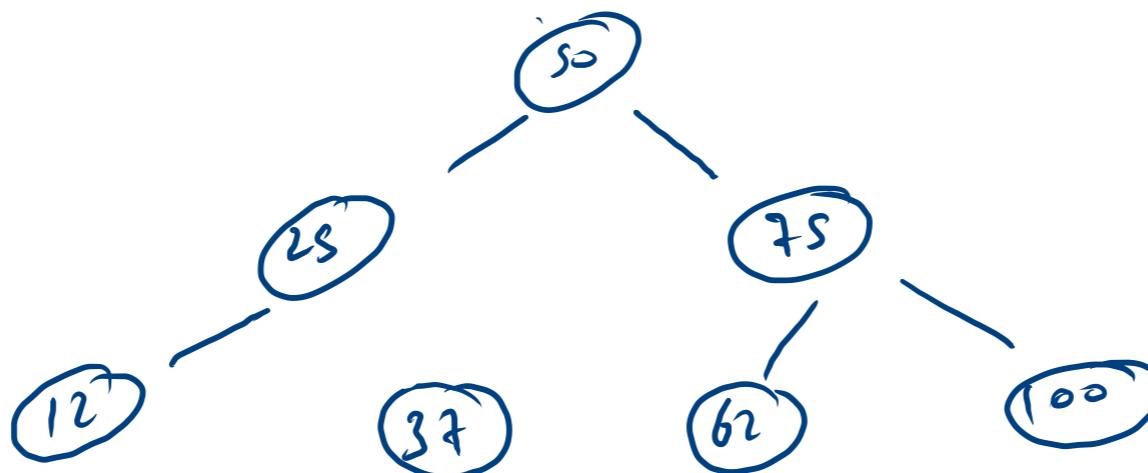
Logic



$n \rightarrow 37$   
 $n \rightarrow 25$   
 $n \rightarrow 50$



remove  $n \leftarrow$   
return root  $\leftarrow$

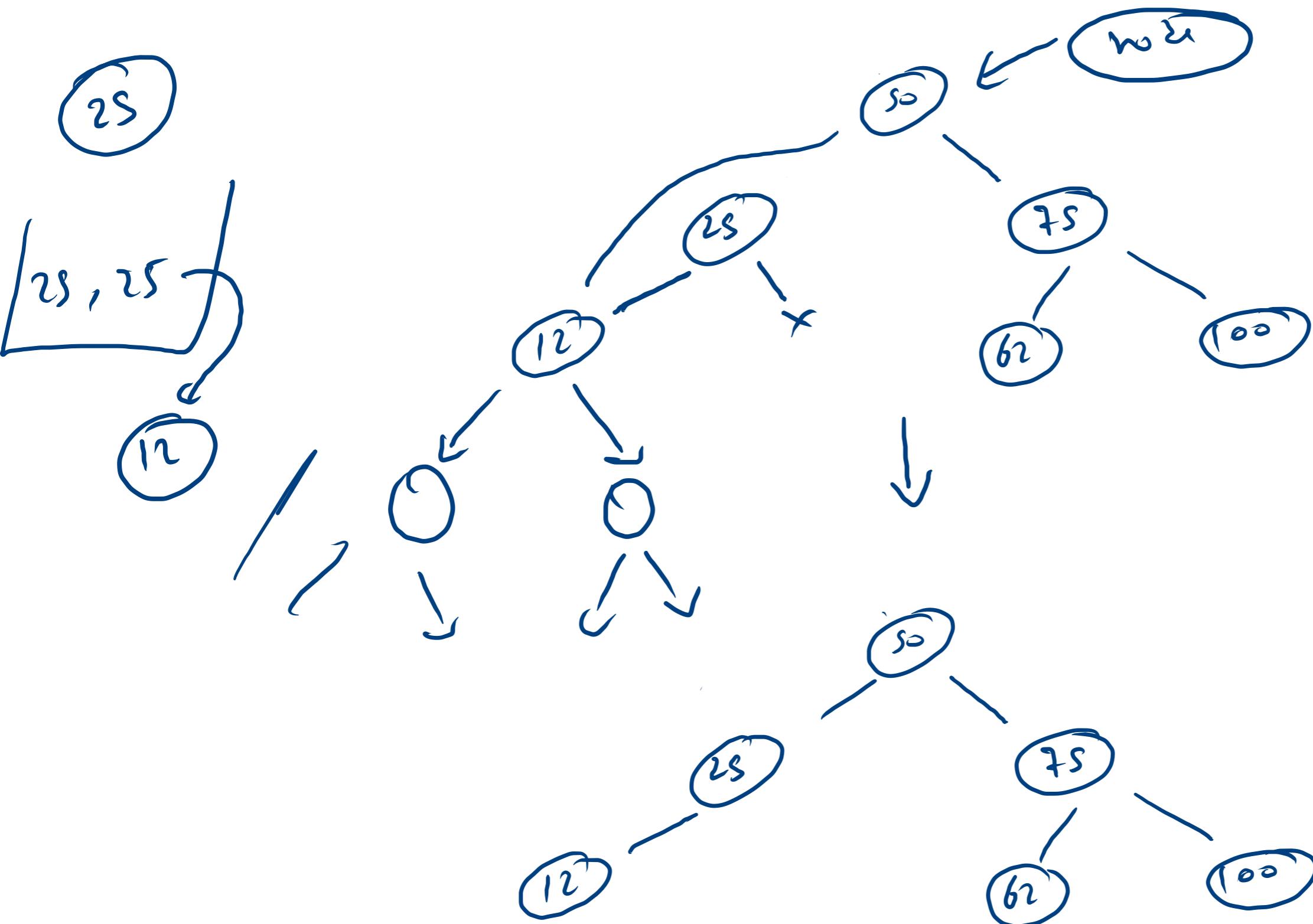


node.left = addL (root.left)

① single child  
root  $\rightarrow$  child

② leaf  
root  $\rightarrow$  null

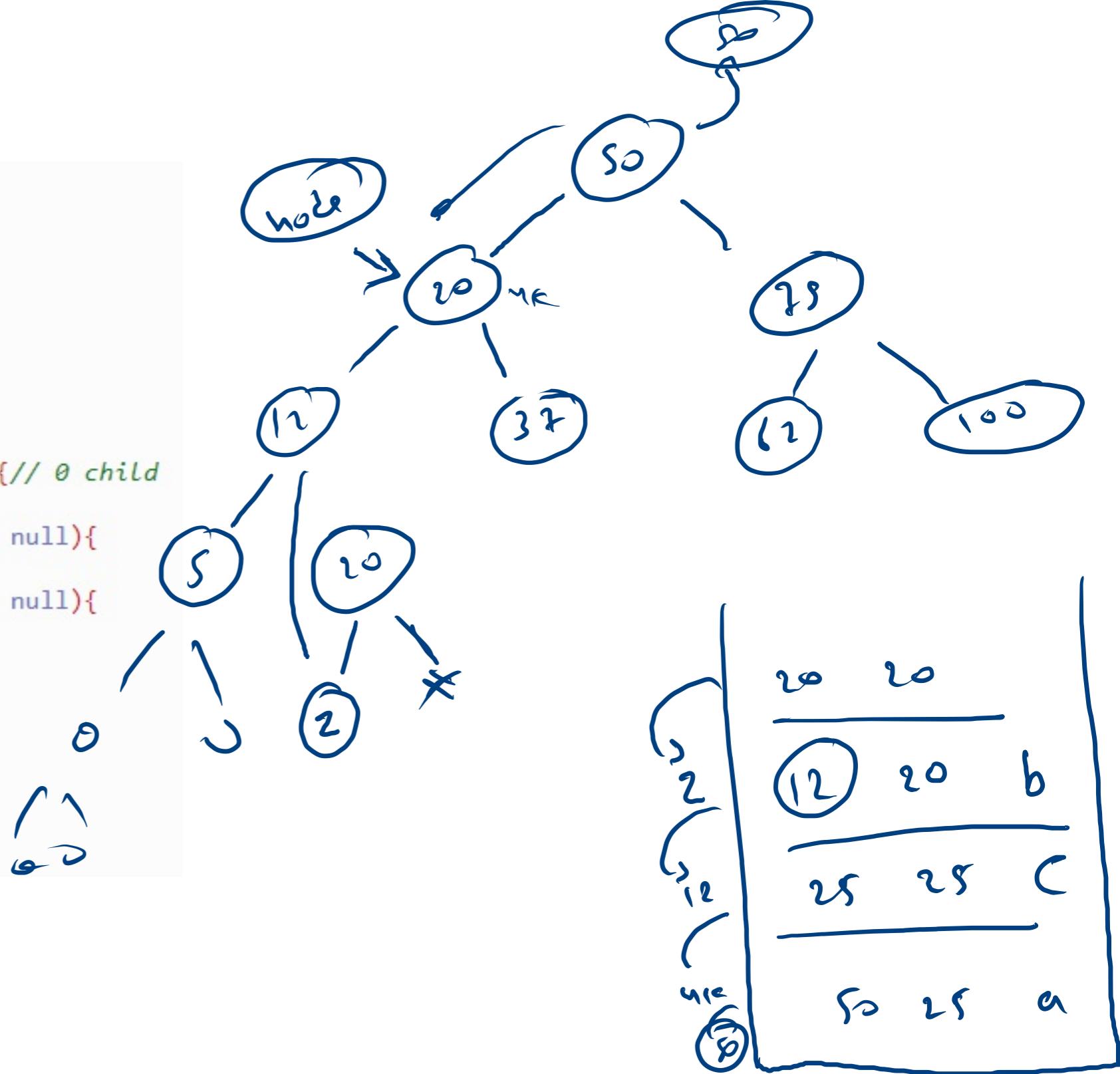
③  
 $n = \text{left max}$   
node.data =  $n$   
remove(left,  $n$ )

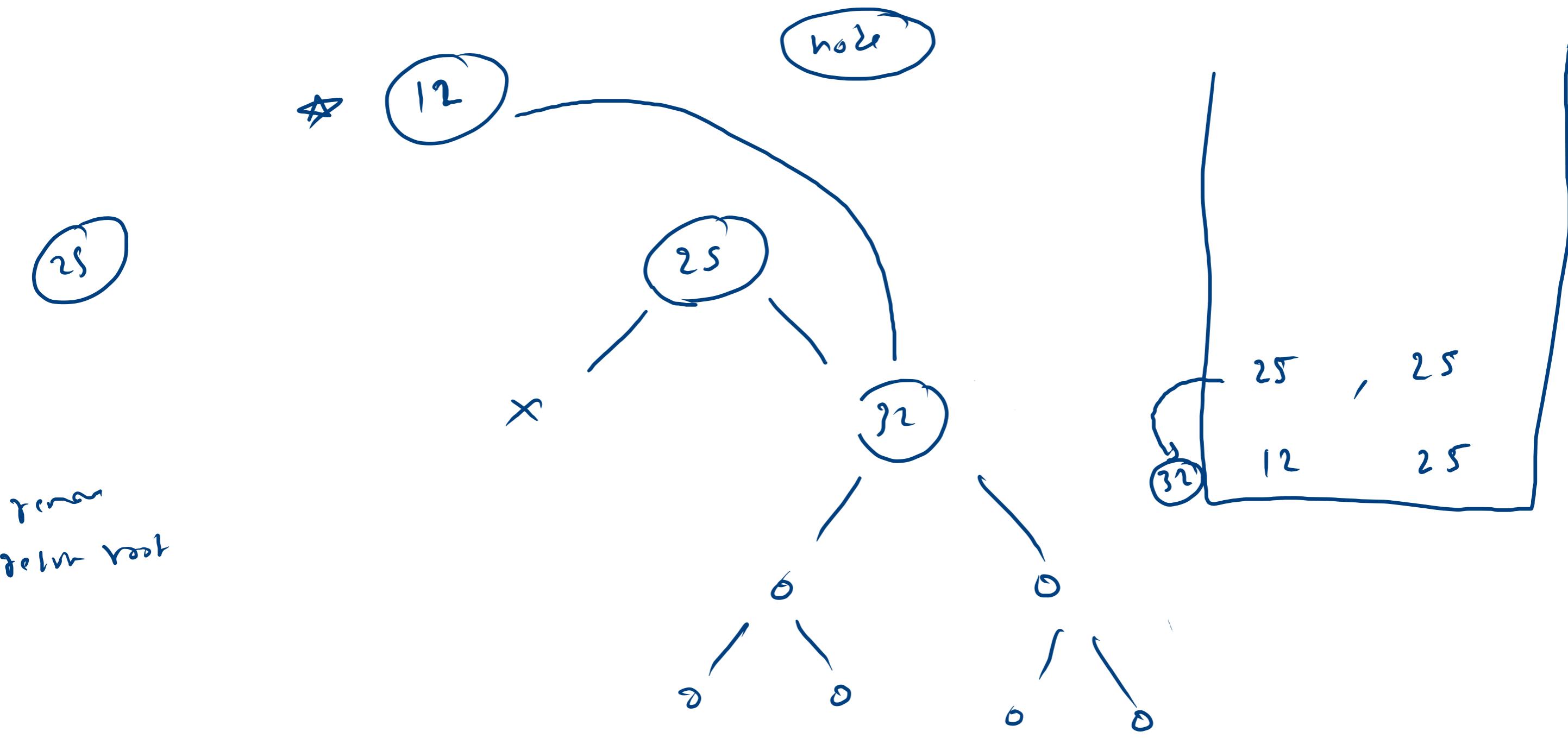


```

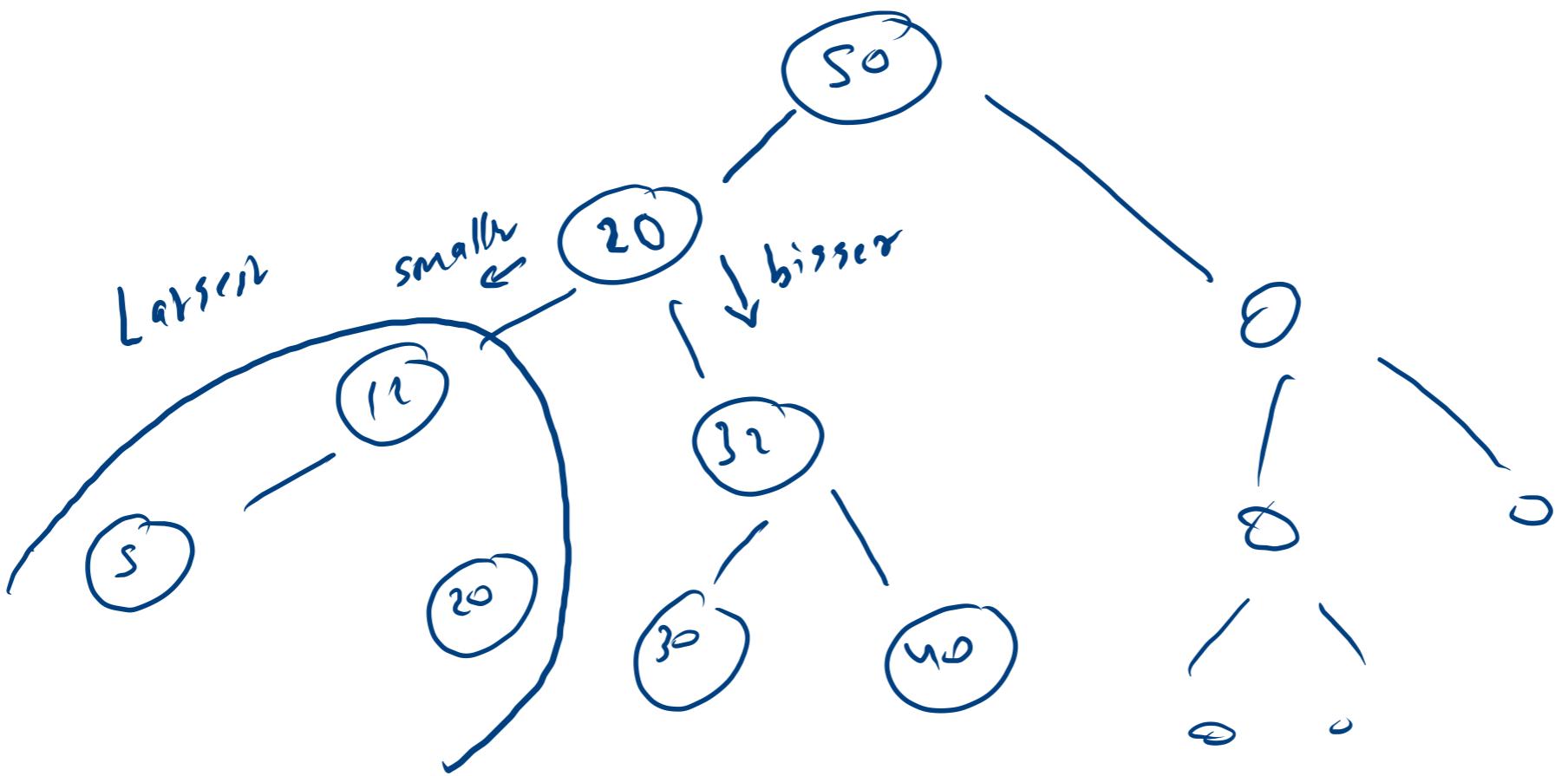
public static Node remove(Node node, int data) {
    if(data < node.data){
        node.left = remove(node.left, data);
        return node;
    }else if(node.data < data){
        node.right = remove(node.right, data);
        return node;
    }else{// data == node.data
        if(node.left == null && node.right == null){// 0 child
            return null;
        }else if(node.left != null && node.right == null){
            return node.left;
        }else if(node.right != null && node.left == null){
            return node.right;
        }else{// 2 child
            int x = max(node.left);
            node.data = x;
            remove(node.left, x);
            return node;
        }
    }
}

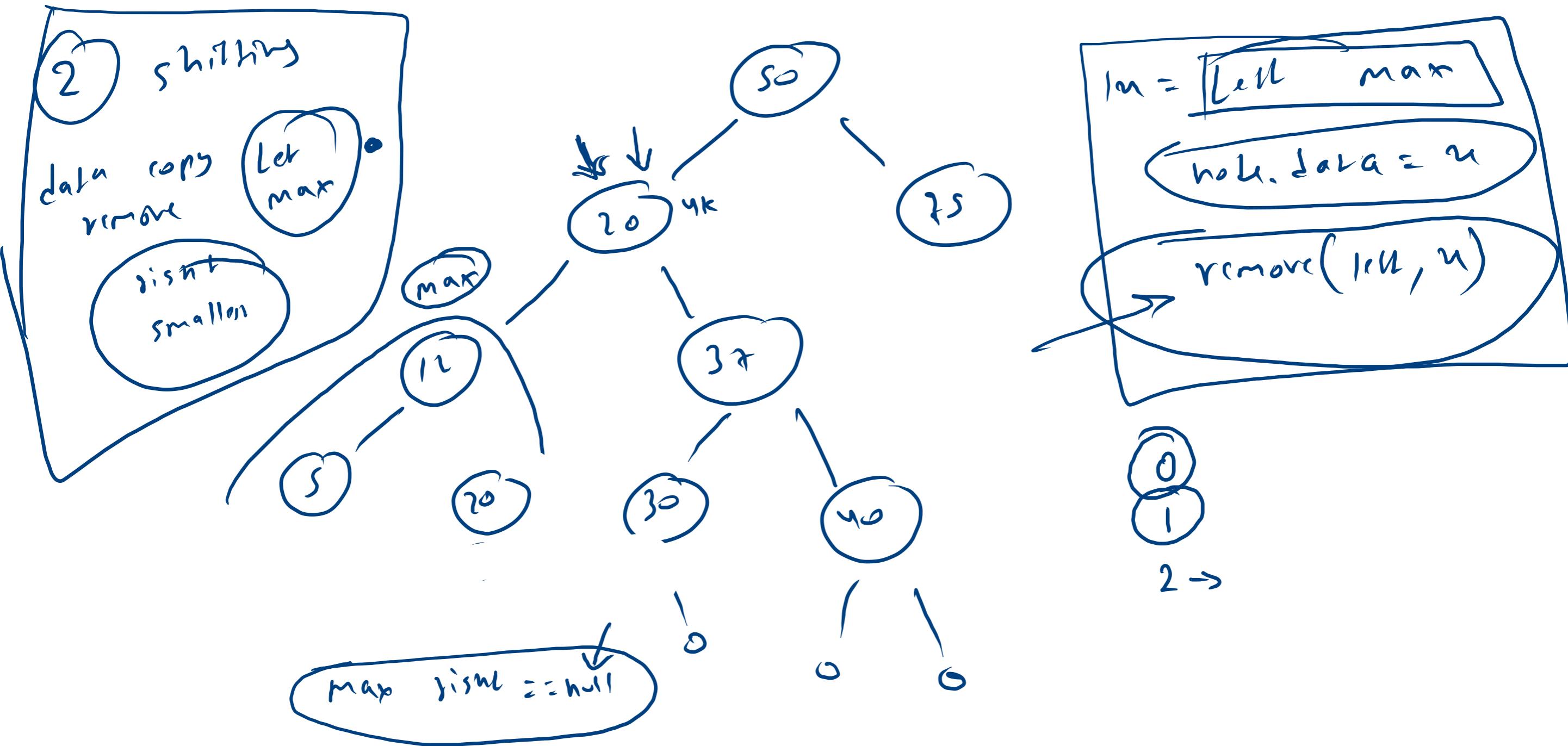
```

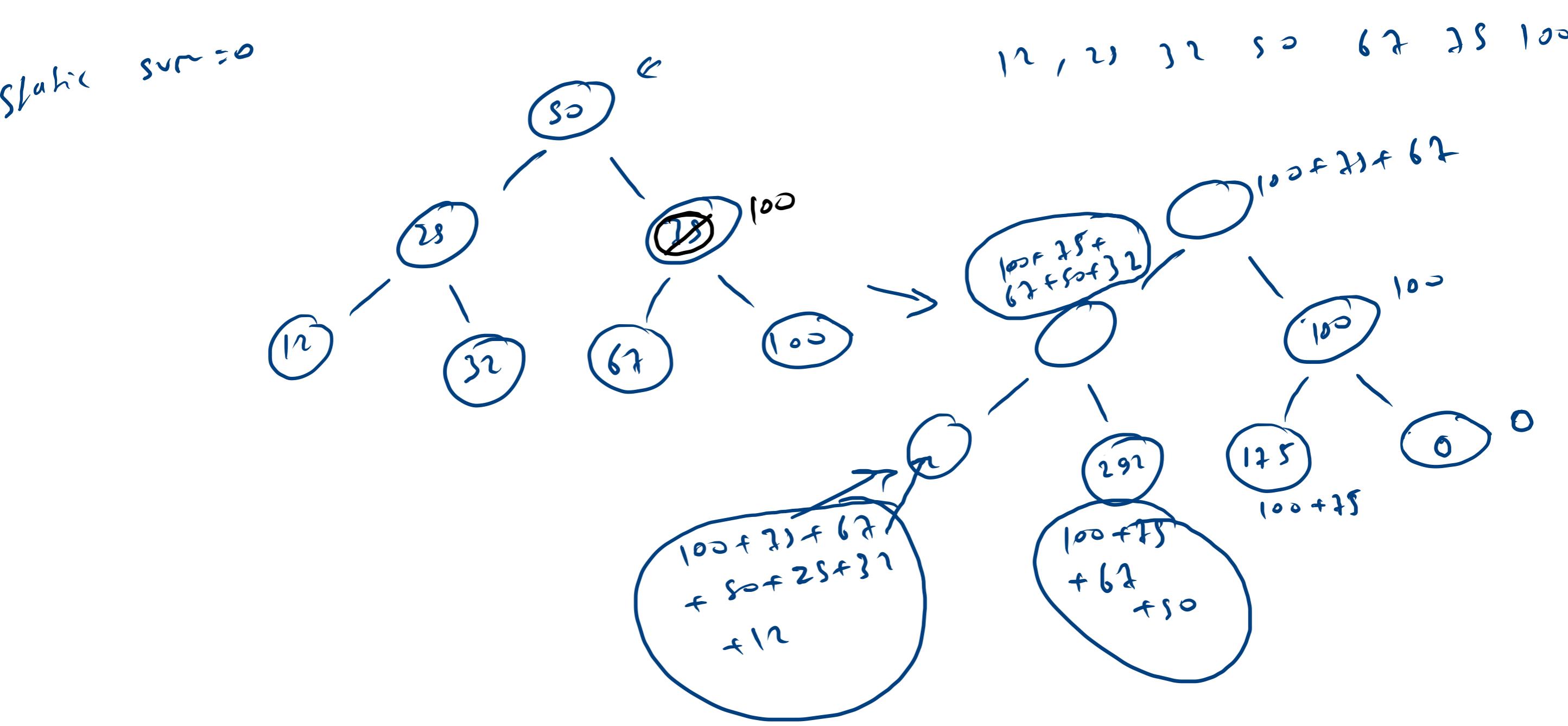


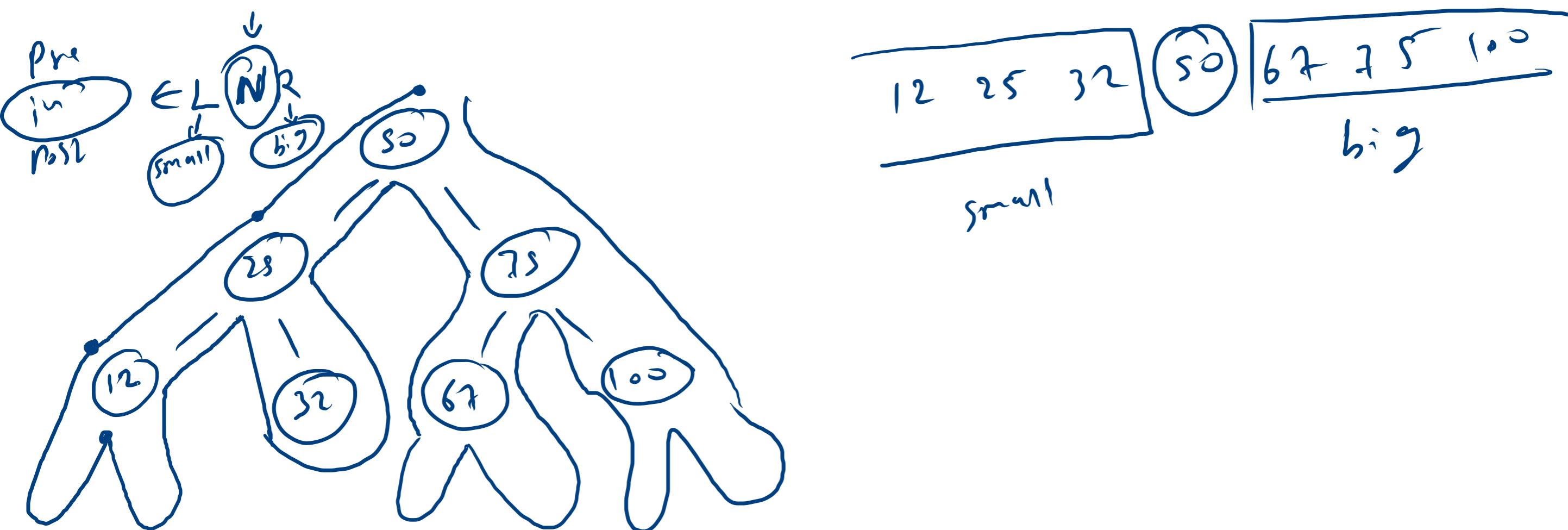


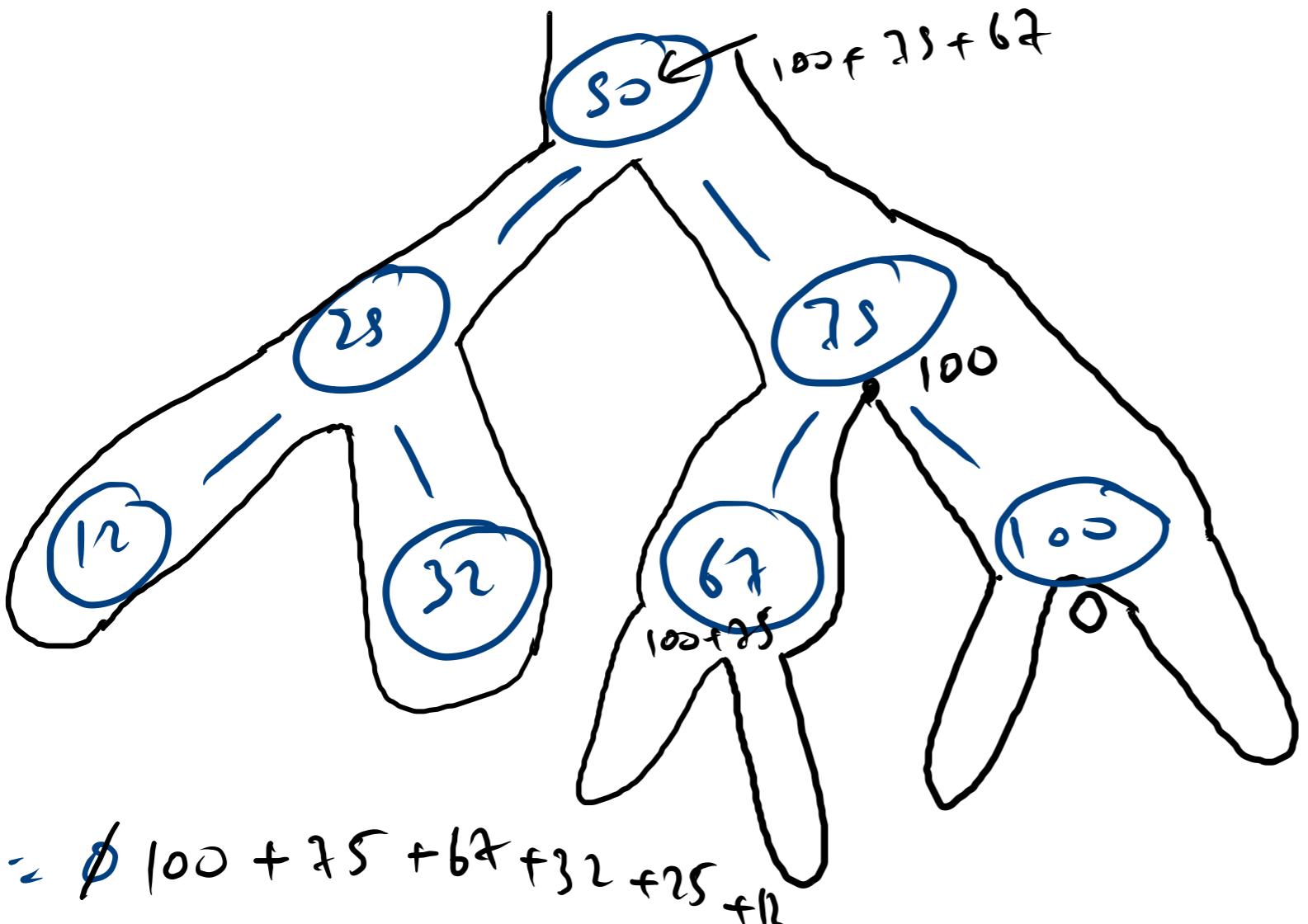
replacement  
remove











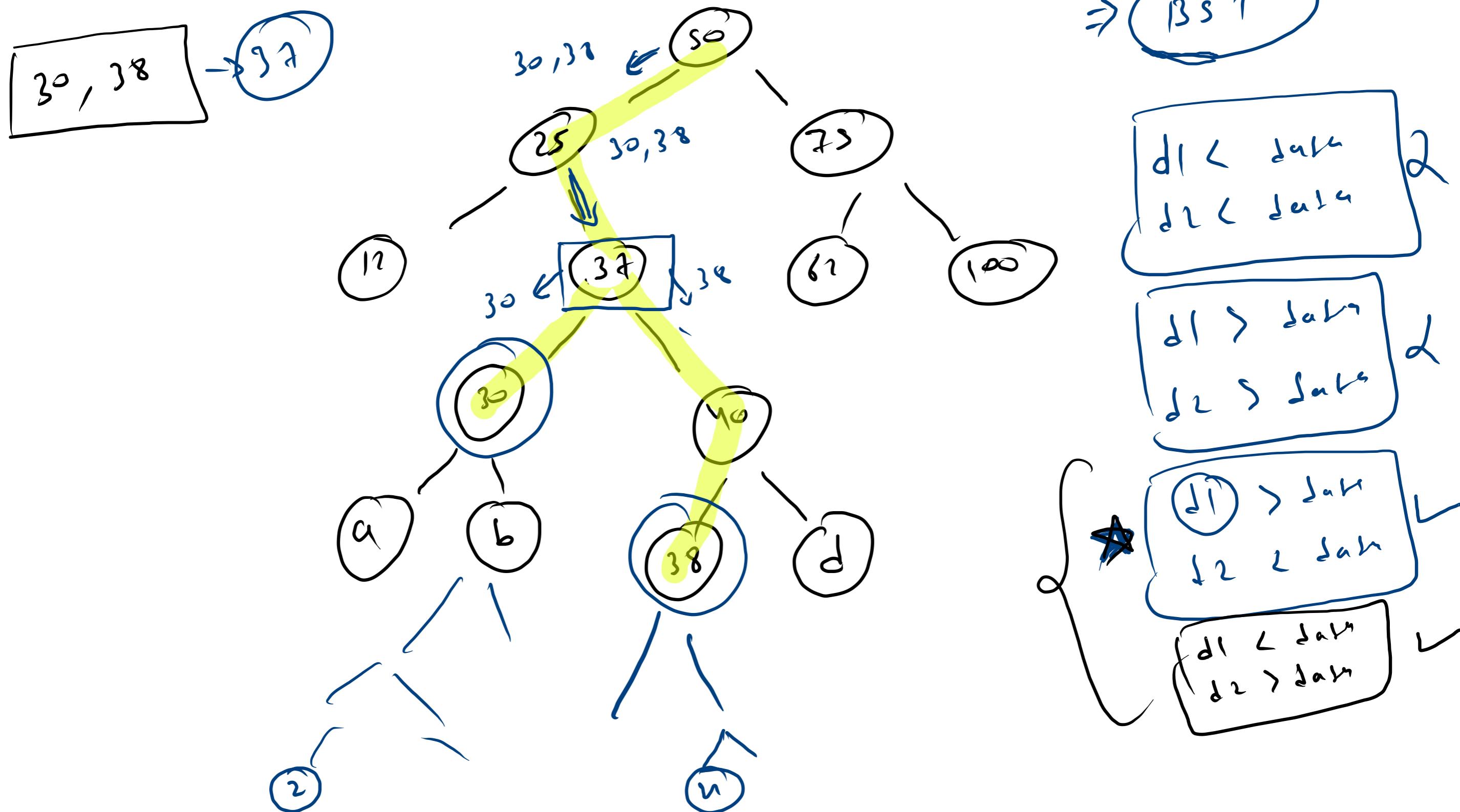
$$\text{sum} = \cancel{0} 100 + 25 + 67 + 32 + 25 + 12$$

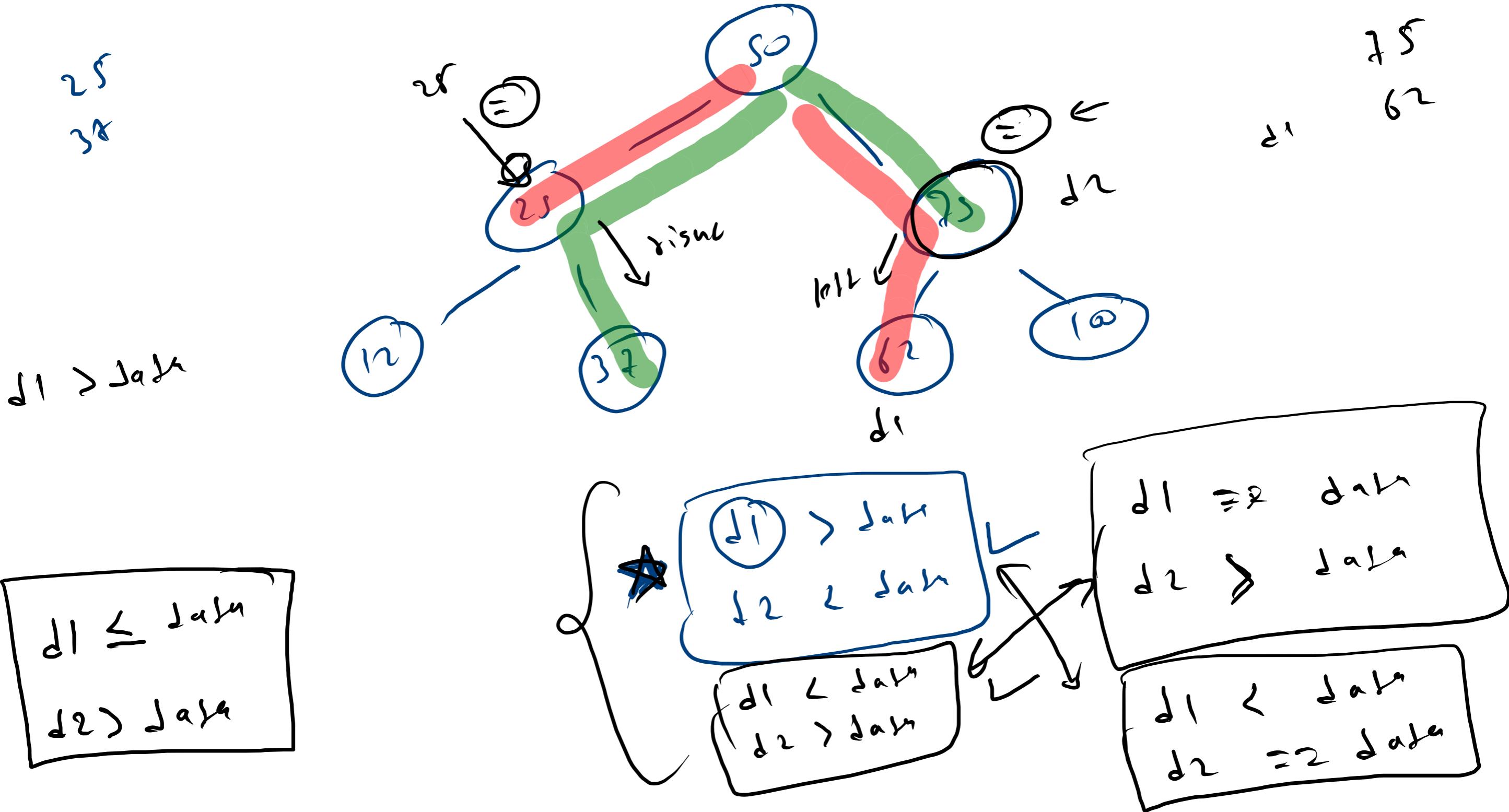
given order  
L N R  
↓  
R ↓ big  
N  
L ↓ small  
faish (right)

node

$n = \text{node.data}$   
 $\text{node.data} = \text{sum}$   
 $\text{sum} += n$   
 faish (left)

so  
big sum





$$d1 = 37$$

$$d2 = 12$$

$$d1 = 12$$

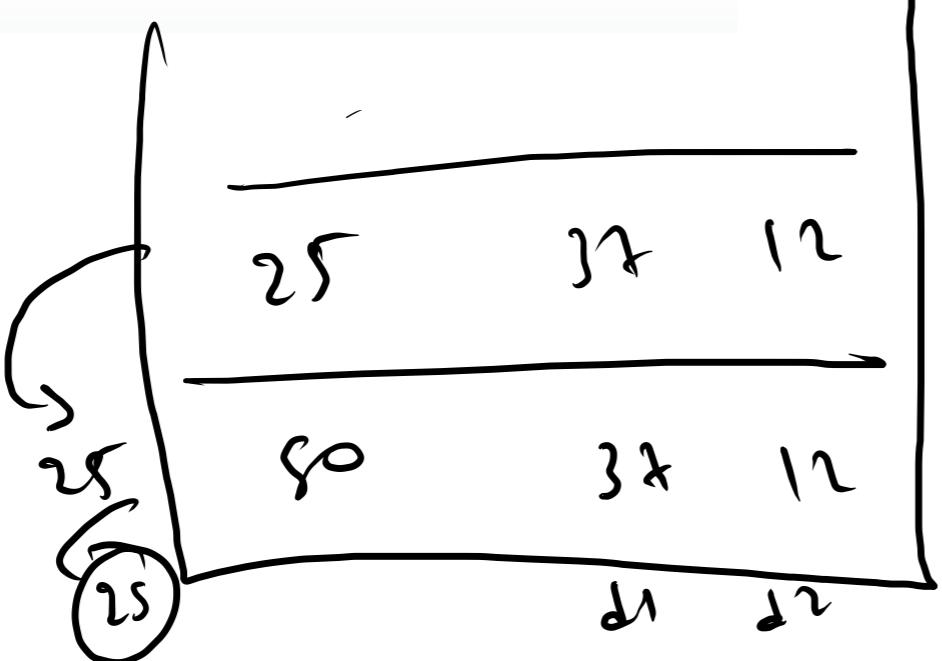
$$d2 = 37$$

```

if(d1 <= node.data && d2 >= node.data){
    return node.data;
}
if(d1 >= node.data && d2 <= node.data){
    return node.data;
}

if(d1 < node.data){
    return lca(node.left, d1, d2);
}
if(d1 > node.data){
    return lca(node.right, d1, d2);
}
return 0;

```



$$n$$

$$d1$$

$$d2$$

