

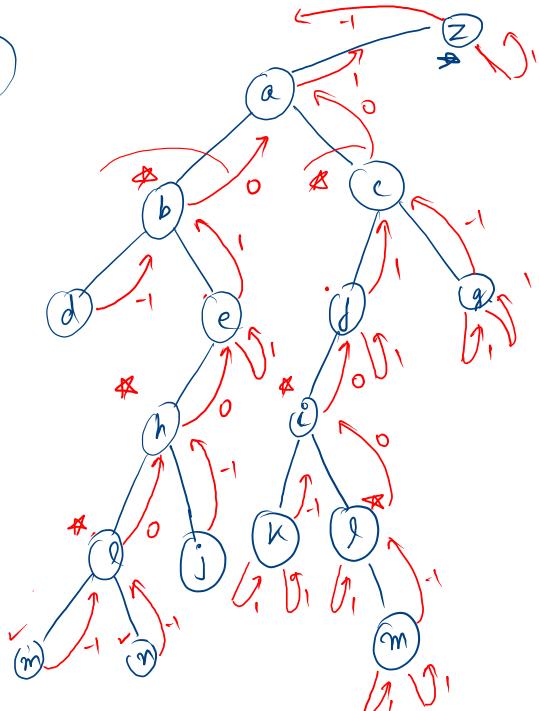
Trees

Spring

Min Camera

Camera : ~~DXL3486~~

- ① algorithmic approach
 - ② Covered
 - ③ need



S Hint

✓
100%

A hand-drawn diagram consisting of two red circles connected by a curved arrow pointing from left to right. The word 'Convergent' is written in red above the first circle. The second circle contains the text 'tend to be convergent'.

- 1 → Camera need
- 0 → Camera placed
- 1 → oxa Currend

```

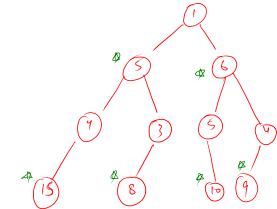
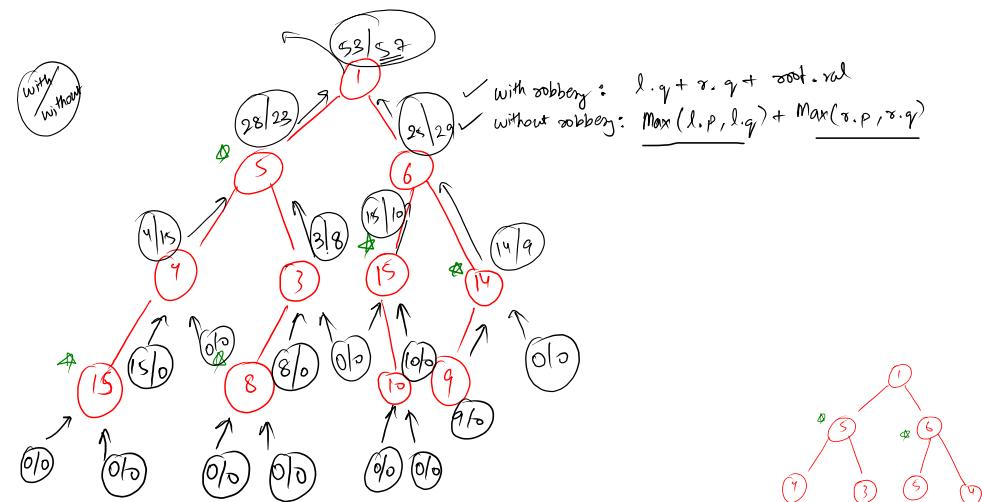
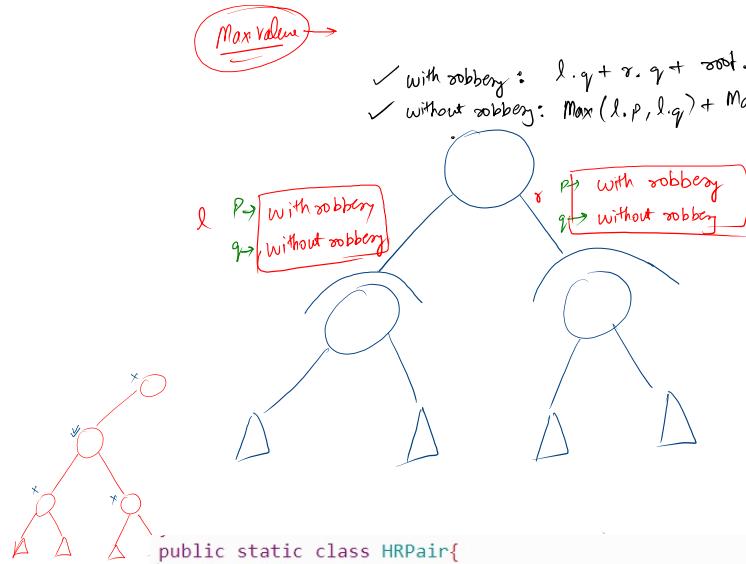
if (lc == -1 || rc == -1)
{
    Camera ++;
    return 0;
}

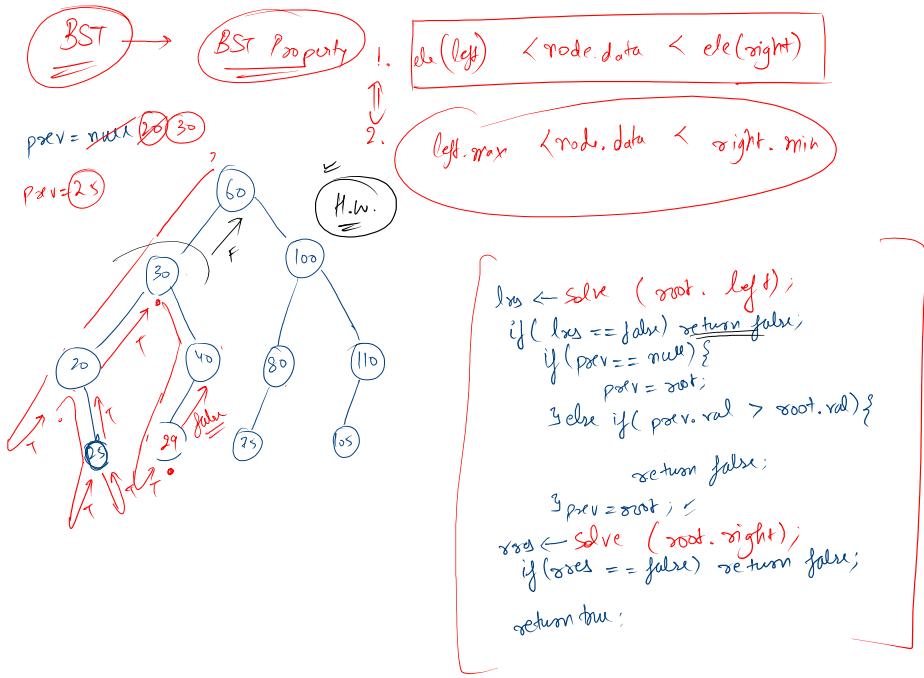
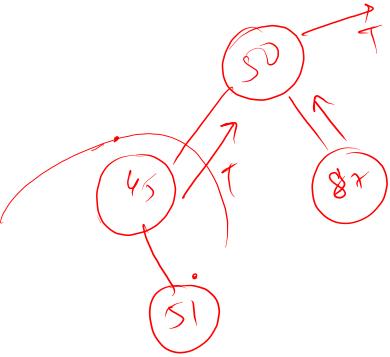
if (lc == 0 || rc == 0)
{
    return 1;
}

return -1;

```

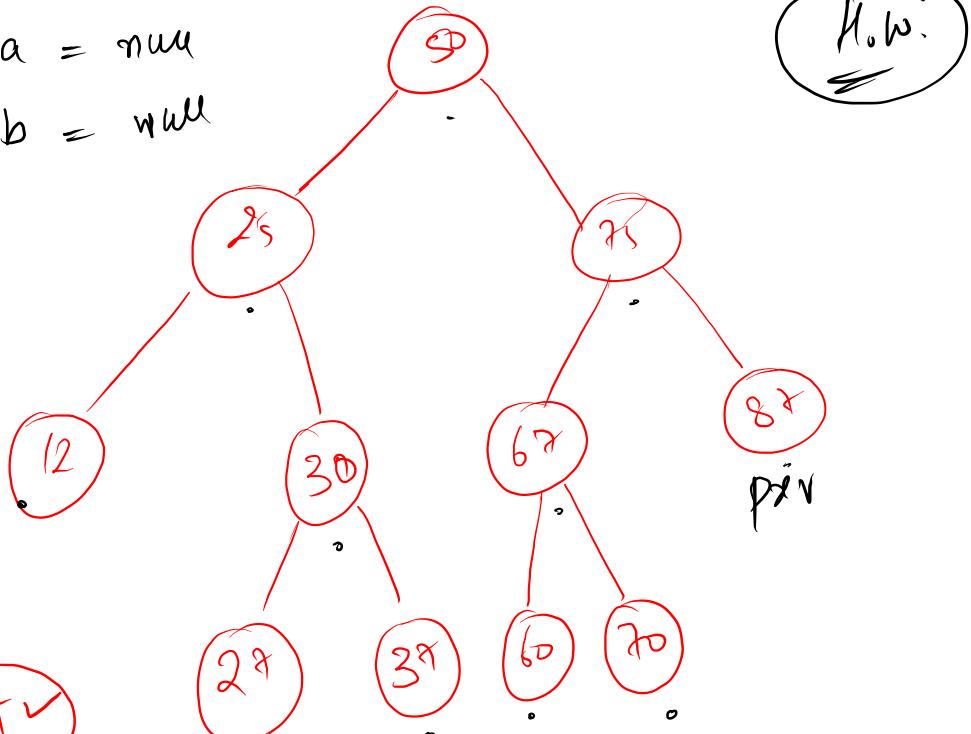
```
public static int MinCamerasInBT(TreeNode root) {  
    camera = 0;  
    if(MinCamerasInBTHelper(root) == -1){  
        camera++;  
    }  
    return camera;  
}  
static int camera;  
private static int MinCamerasInBTHelper(TreeNode root){  
    if(root == null){  
        return 1;  
    }  
  
    int ls = MinCamerasInBTHelper(root.left);  
    int rs = MinCamerasInBTHelper(root.right);  
  
    if(ls == -1 || rs == -1){  
        camera++;  
        return 0;  
    }  
    if(ls == 0 || rs == 0){  
        return 1;  
    }  
    return -1;  
}
```





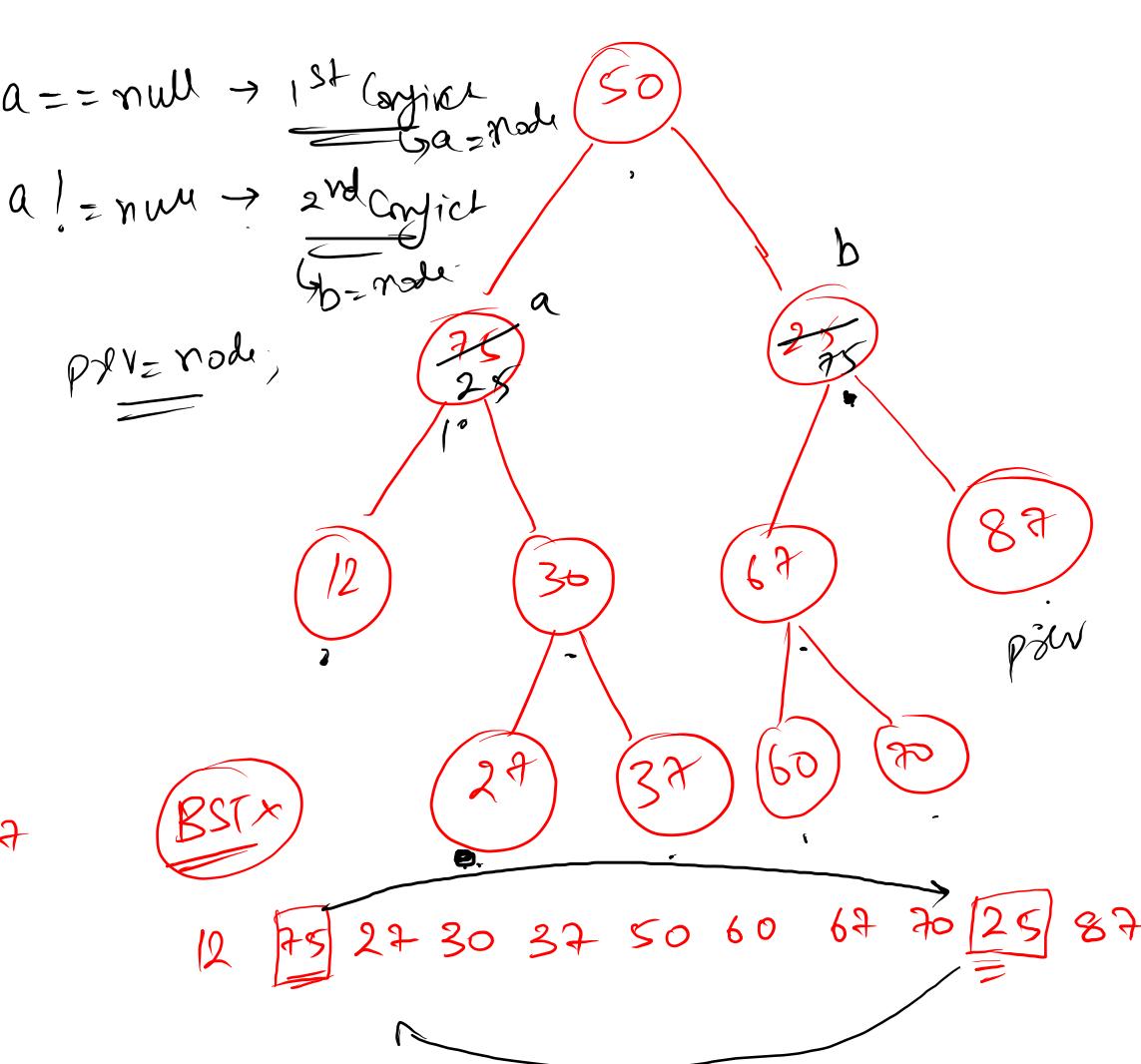
$a = \text{null}$

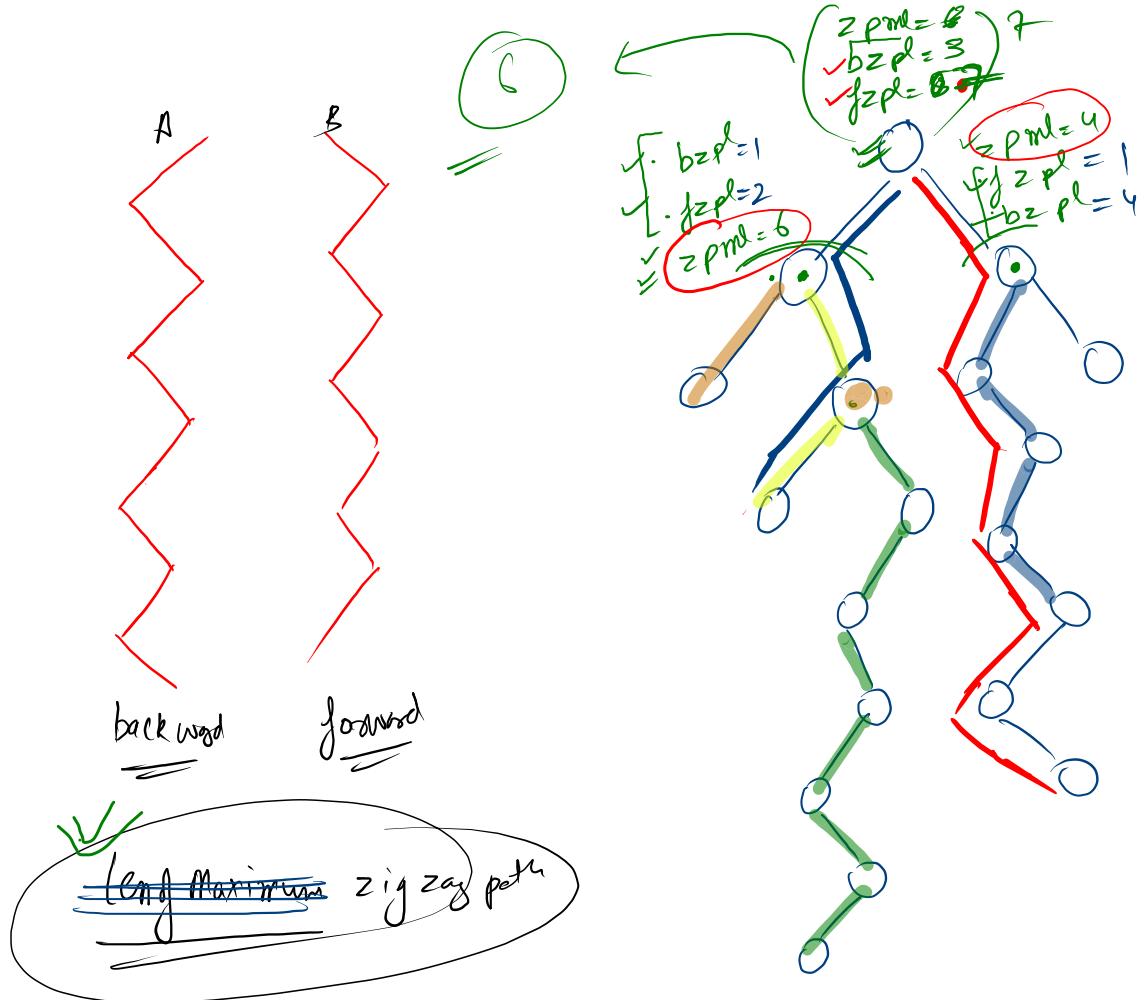
$b = \text{null}$



~~BST ✓~~

12 25 28 30 34 50 60 62 70 75 87

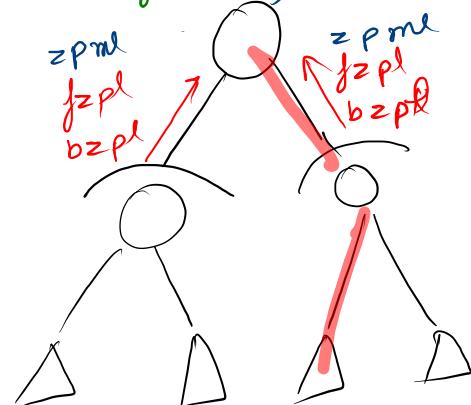


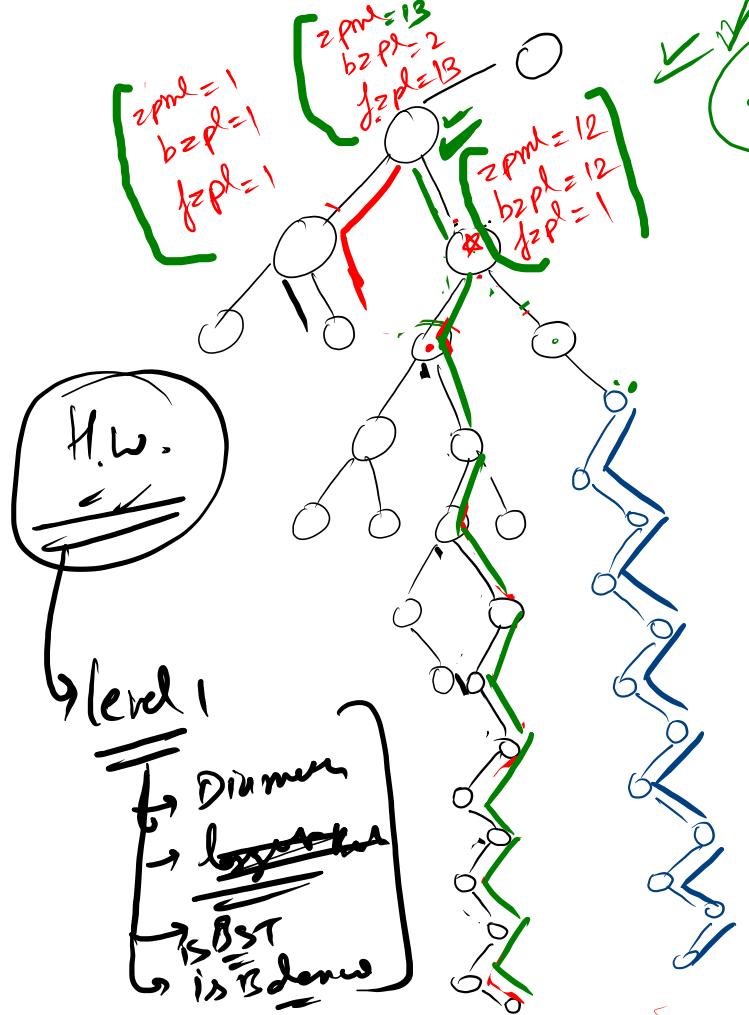


$\text{zpmle} = \max(\underline{\text{left.zpmle}}, \underline{\text{right.zpmle}}, \underline{\text{bzpl}}, \underline{\text{fzpl}});$

$\underline{\text{bzpl}} = \text{left.fzpl} + 1$

$\underline{\text{fzpl}} = \text{right.bzpl} + 1$





$$zPath = \max(b2pl, j2pl, l.zPath, r.zPath)$$

2 13 1 12

$\cdot b2pl$] z-path from root
 $\cdot j2pl$]

$\rightarrow \text{left.zPath}$] max len of z-path from
 lefttree (always not from
 leftchild)

$\rightarrow \text{right.zPath}$] =

$=$
 $=$
 $-$
 $-$
 $-$
 $-$

f

