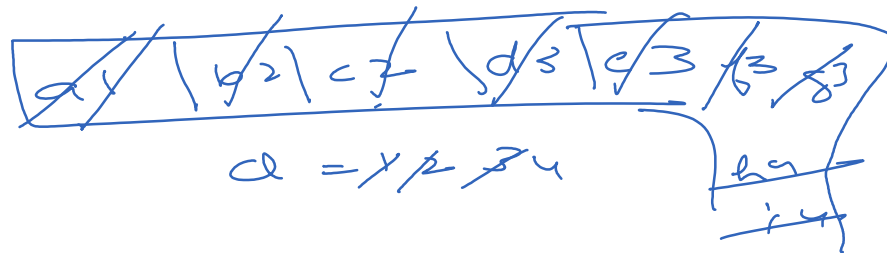
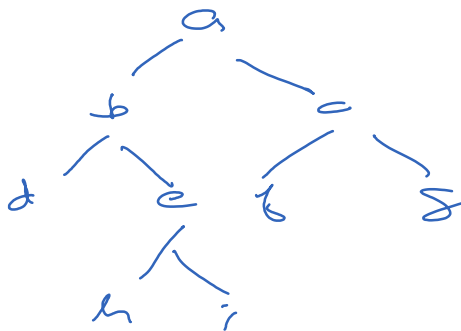


9:25

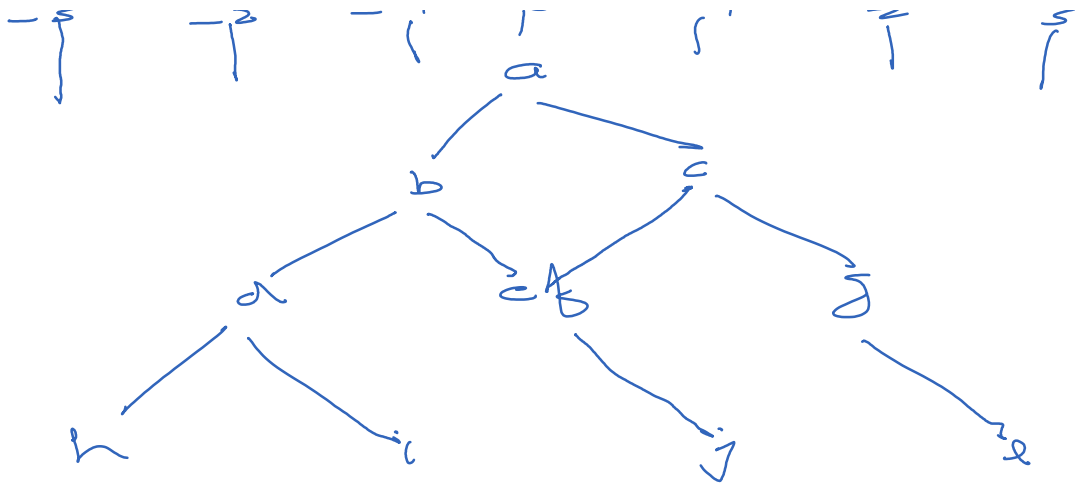
lect

$\langle h, i \rangle$   
 $\langle d, e, f, g \rangle$   
 $\langle b, c \rangle$   
 $\langle a \rangle$

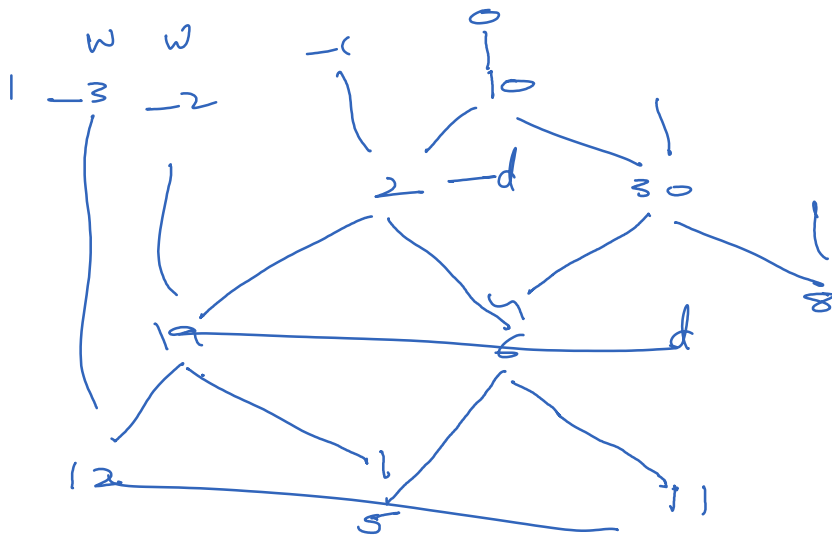


$\langle a \rangle$   
 $\langle b, c \rangle$   
 $\langle d, e, f, g \rangle$   
 $\langle h, i \rangle$

$-3$     $-2$     $-1$     $0$     $1$     $2$     $3$   
 $-a$     $-$     $-$     $-$     $-$     $-$     $-$



$[h]$   
 $[d]$   
 $[b, i]$   
 $[a, c, f]$   
 $[e, j]$   
 $[g]$   
 $[k]$



$[12]$

$[19]$

$[2, 1, 5]$

$[10, 4, 6]$

$[30, 11]$

$[8]$

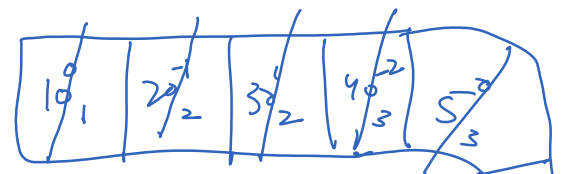
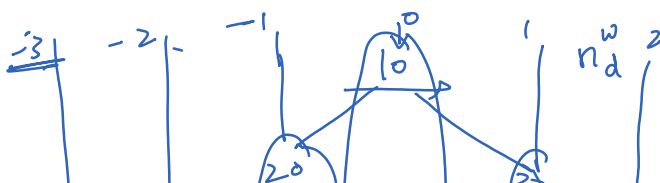
$\Rightarrow$

Confer

$HM \langle J, AL \langle P \rangle \rangle$

$-3 \quad [12^{-3}_9] \quad -2 \quad [19^{-2}_3]$

$-1 \quad [2^{-1}_2 \quad 4^{-1}_4 \quad 5^{-1}_5]$





And Your Code's output is: ☐

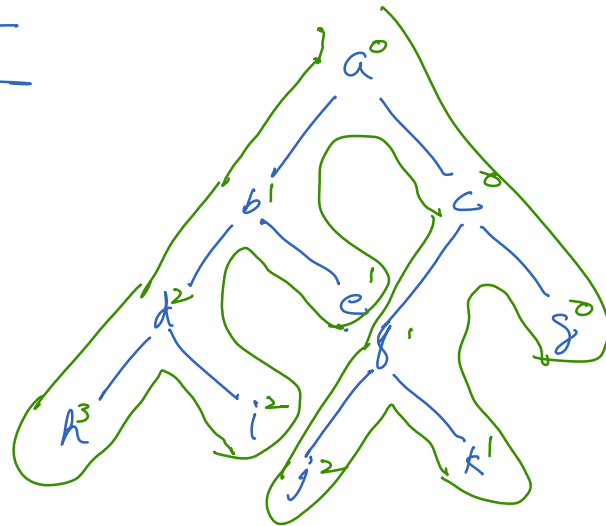
12 12 14 7 11 1 12 3 4 12 8 7 7 6 2 10 11 14 7 14

Its Correct output is: ☐

12 12 14 7 11 1 3 8 12 4 12 7 7 2 10 6 11 14 7 14

0 → a, c, g  
1 → b, e, f, i  
2 → d, j, k  
3 → h

a c g, b e f i, d j k, h



```
static void traversal(Node root, int diag, l
{
    if(root == null){
        return;
    }

    if(diag > md){
        md = diag;
    }

    if(map.containsKey(diag) == false){
        map.put(diag, new ArrayList<>());
        map.get(diag).add(root.data);
    } else {
        map.get(diag).add(root.data);
    }

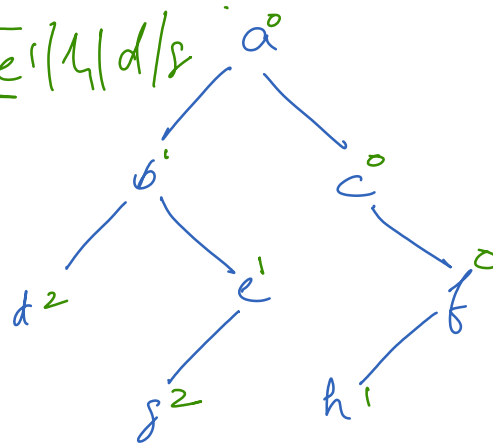
    traversal(root.left, diag + 1, map);
    traversal(root.right, diag, map);
}
```

```
public ArrayList<Integer> diagonal(Node root)
{
    HashMap<Integer, ArrayList<Integer>> map = new HashMap<>();
    traversal(root, 0, map);

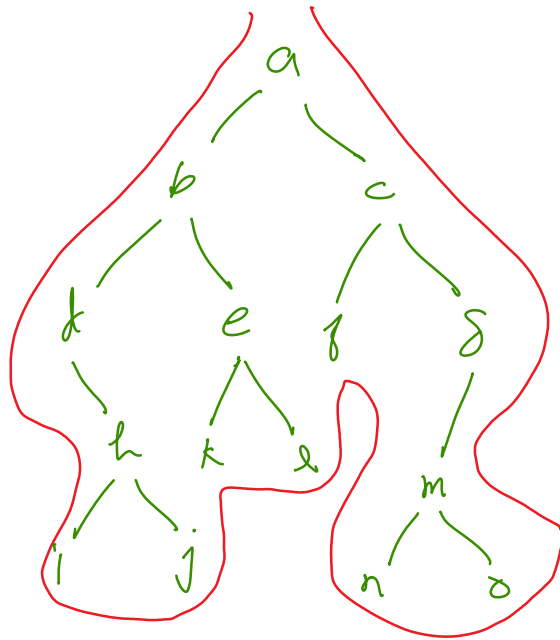
    ArrayList<Integer> total = new ArrayList<>();
    for(int d = 0; d <= md; d++){
        ArrayList<Integer> list = map.get(d);
        for(int item: list){
            total.add(item);
        }
    }

    return total;
}
```

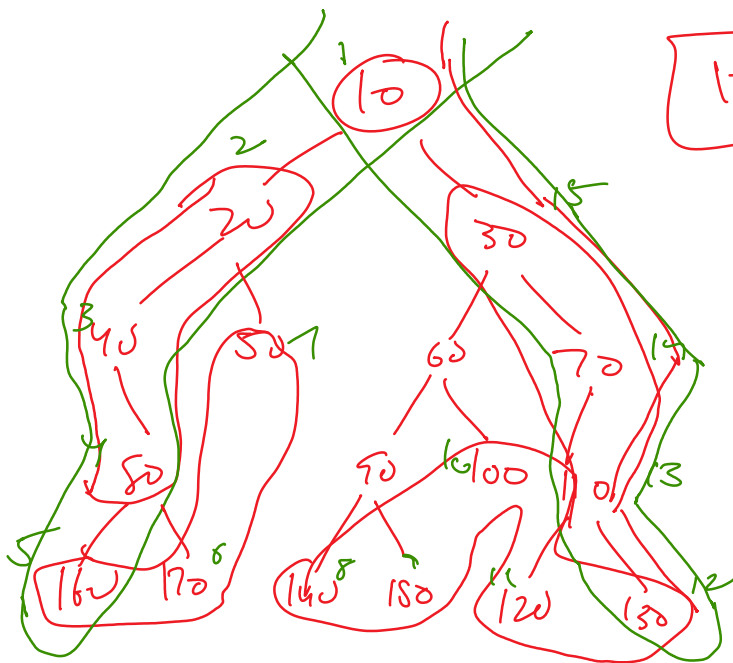
rus a0 | c0 | g0 | b1 | e1 | h1 | d2 | f2



a0 | b1 | h1 | d2 | f2



2ω pro  
 a l  
 2ω posto



10 | 20 | 40 | 80 | 160 | 170 | 58