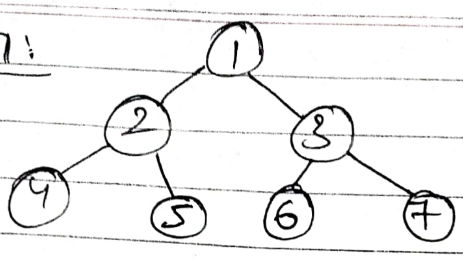
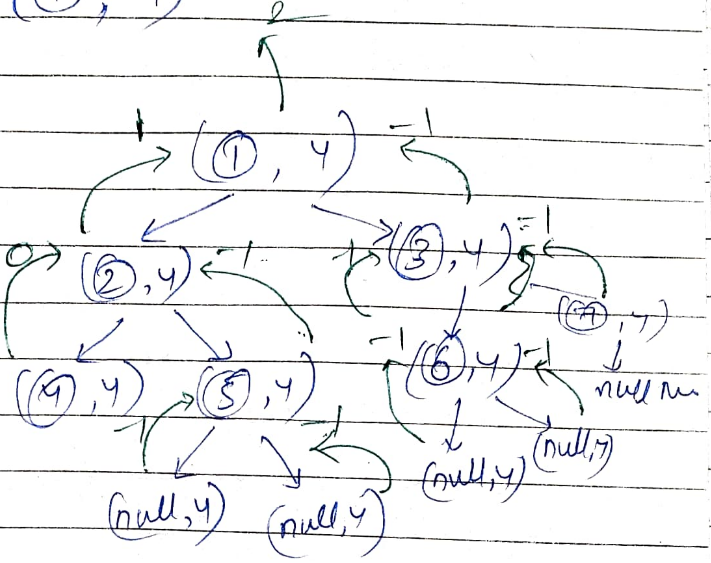
Dry Run:



$n_1 = 4$ $n_2 = 6$

lca = ①

dist1 = lcaDist(①, 4)



~~lcaDist(null, 4)~~

~~lcaDist(null, 4)~~

lca Dist(⑤, 4) -1 -1

~~lcaDist(④, 4)~~

lcaDist(②, 4) 0 =1

lcaDist(①, 4)

right dist = -1

~~lca(null, 4)~~

leftDist = -1

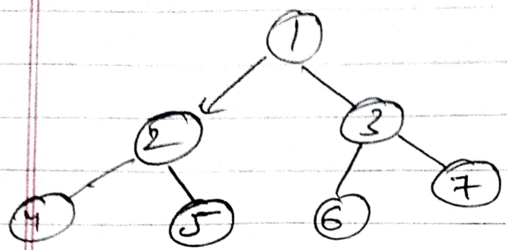
~~lca(null, 4)~~

leftDist → lca(④, 4) 1

leftdist → lca(②, 4)

~~dist~~ lcaDist(①, 4)

Q. k^{th} Ancestor of a Node



n=4 k=2

```
public static int kAncestor(Node root, int data, int k){
    if(root == null) return -1;
    if(root.data == data) return 0;
    int leftDistance = kAncestor(root.left, data, k);
    int rightDistance = kAncestor(root.right, data,k);
    if((leftDistance ==-1 && rightDistance ==-1)
              return -1;
else{
      int Max = Math.max(leftDistance, rightDistance);
      if(Max+1 ==k) Sout (root.data);
      return max+1;
}
```