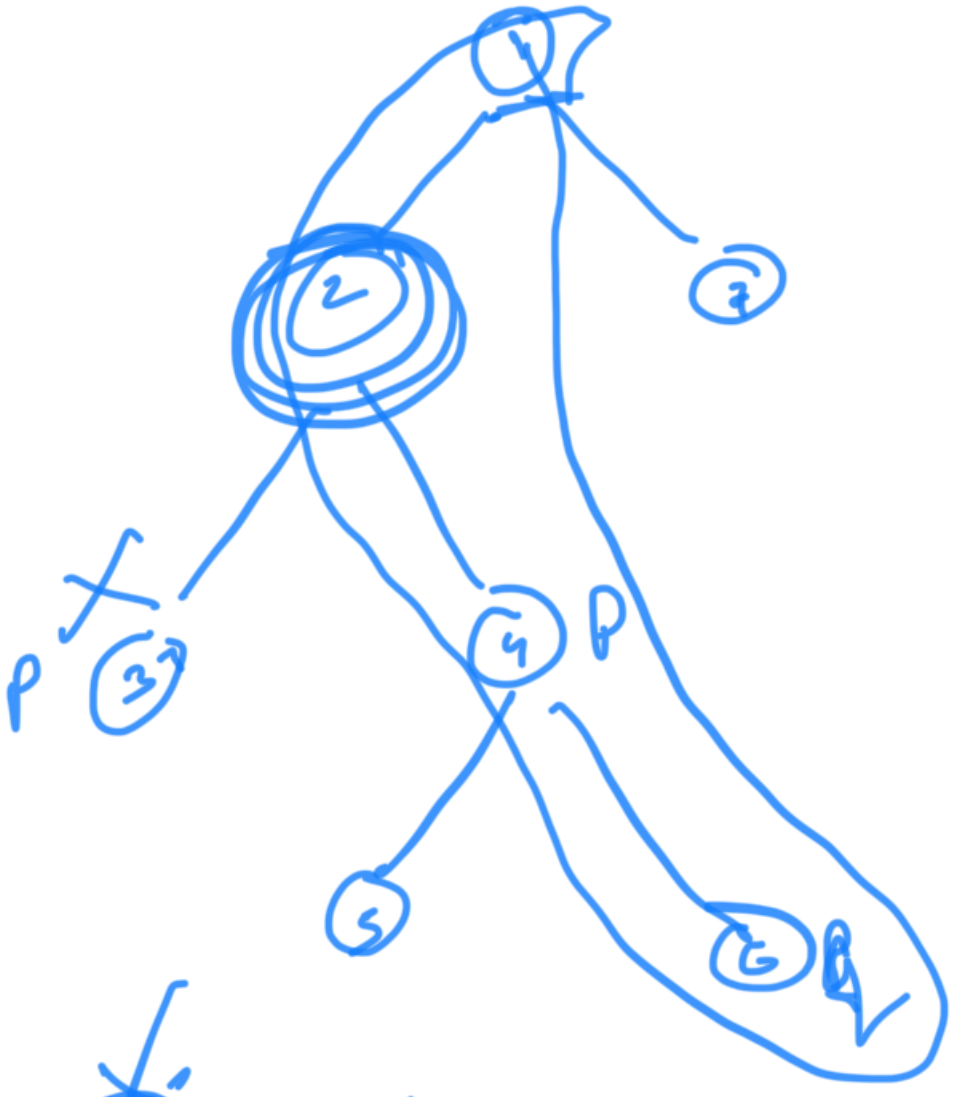
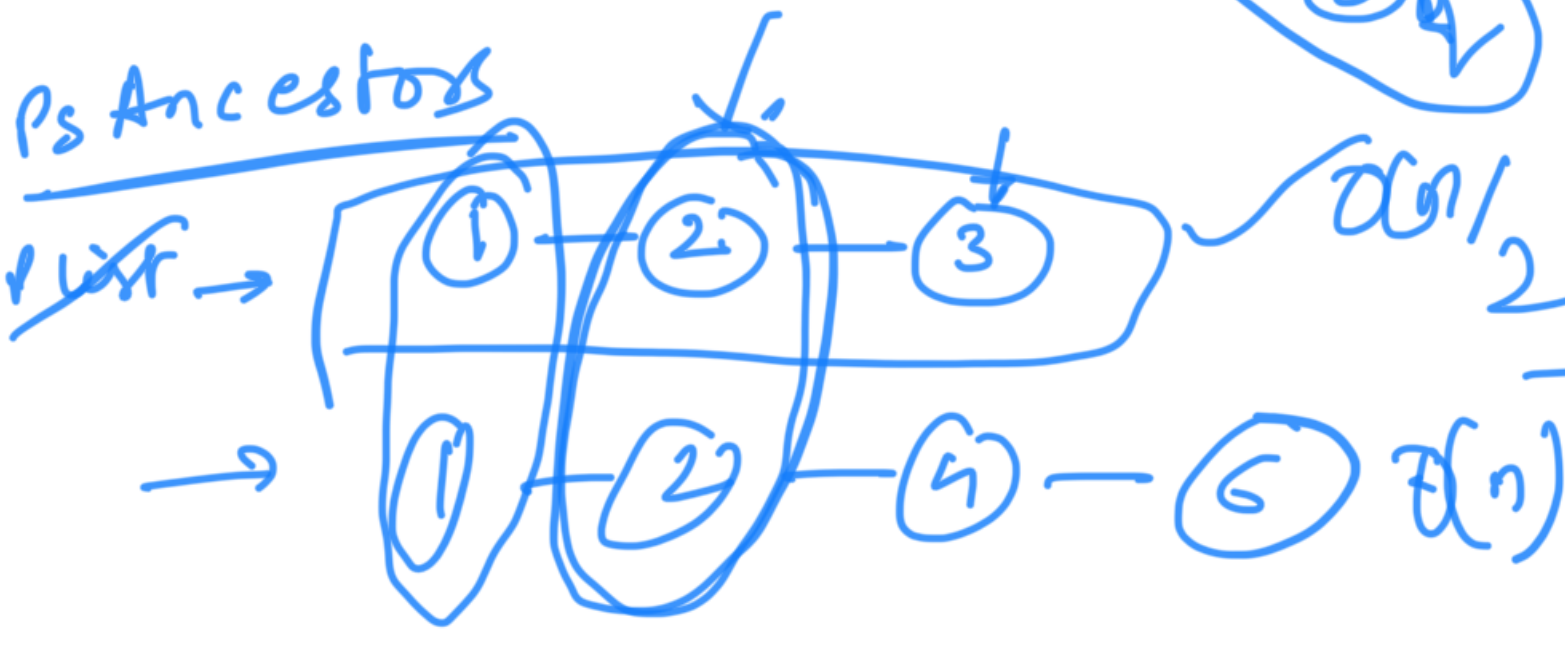


1 - 2 - 3 - 4

O(n)

O(n)

Ps Ancestors

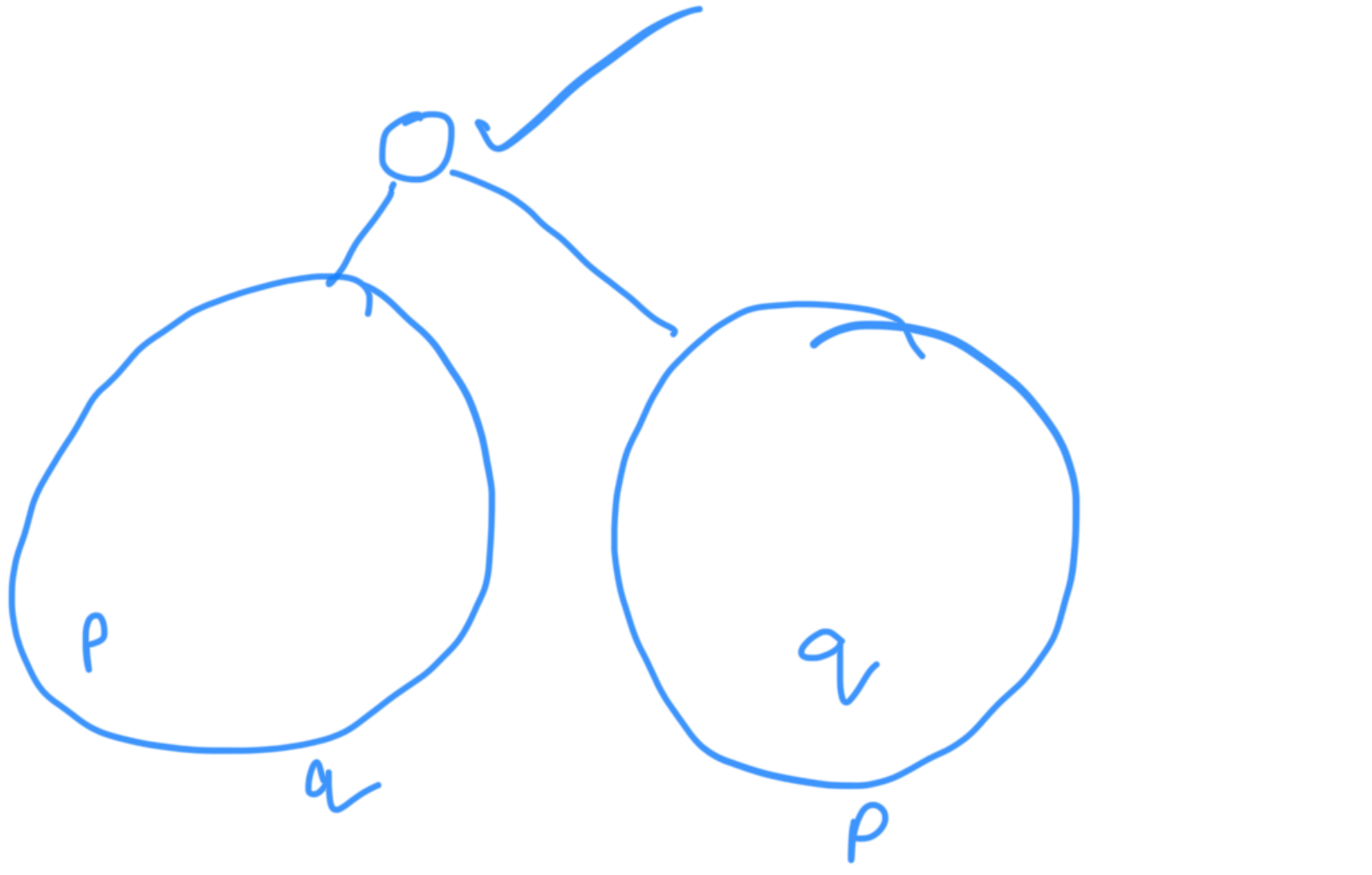


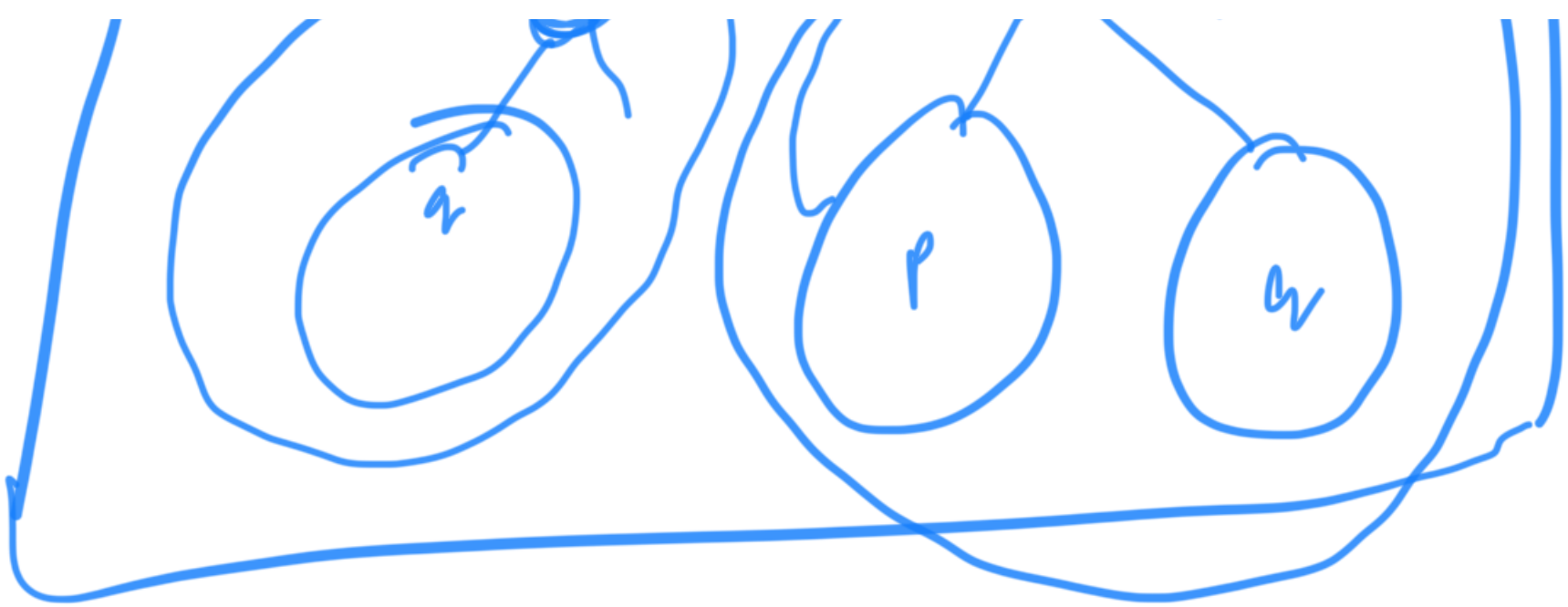
O(n/2

O(n)

Linked list

post order traversal

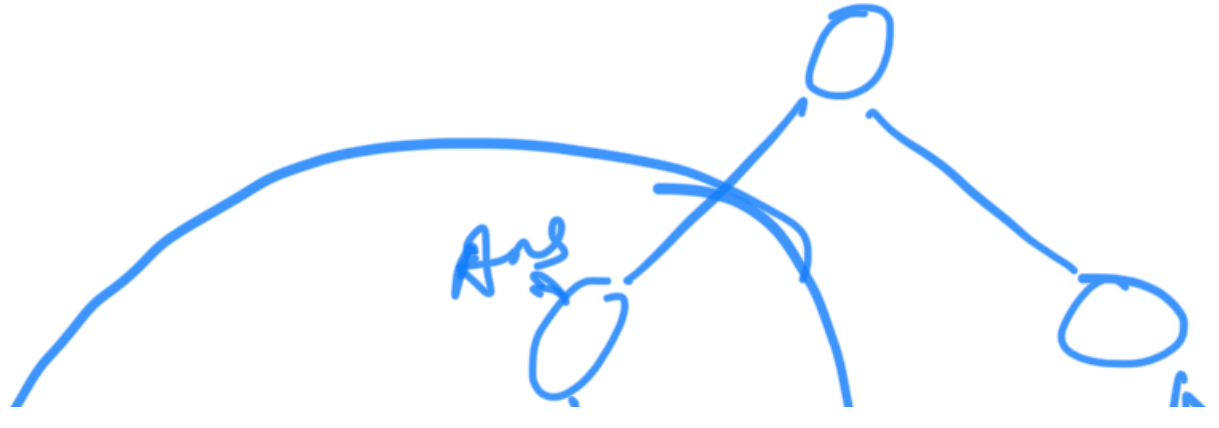


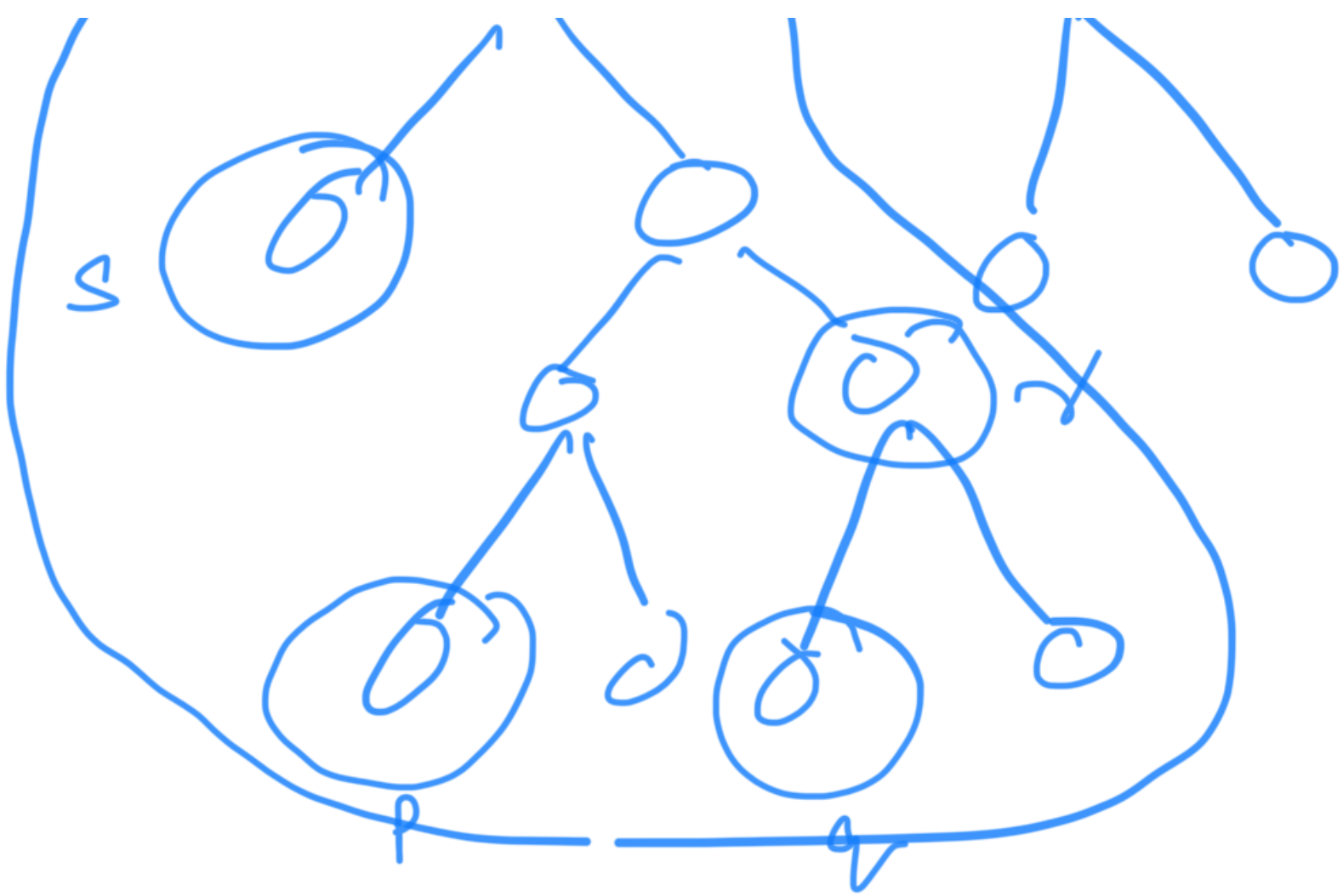


what if it is a BST

what if it an n-array tree

what if instead of 2 nodes p and q
a list of k nodes is given



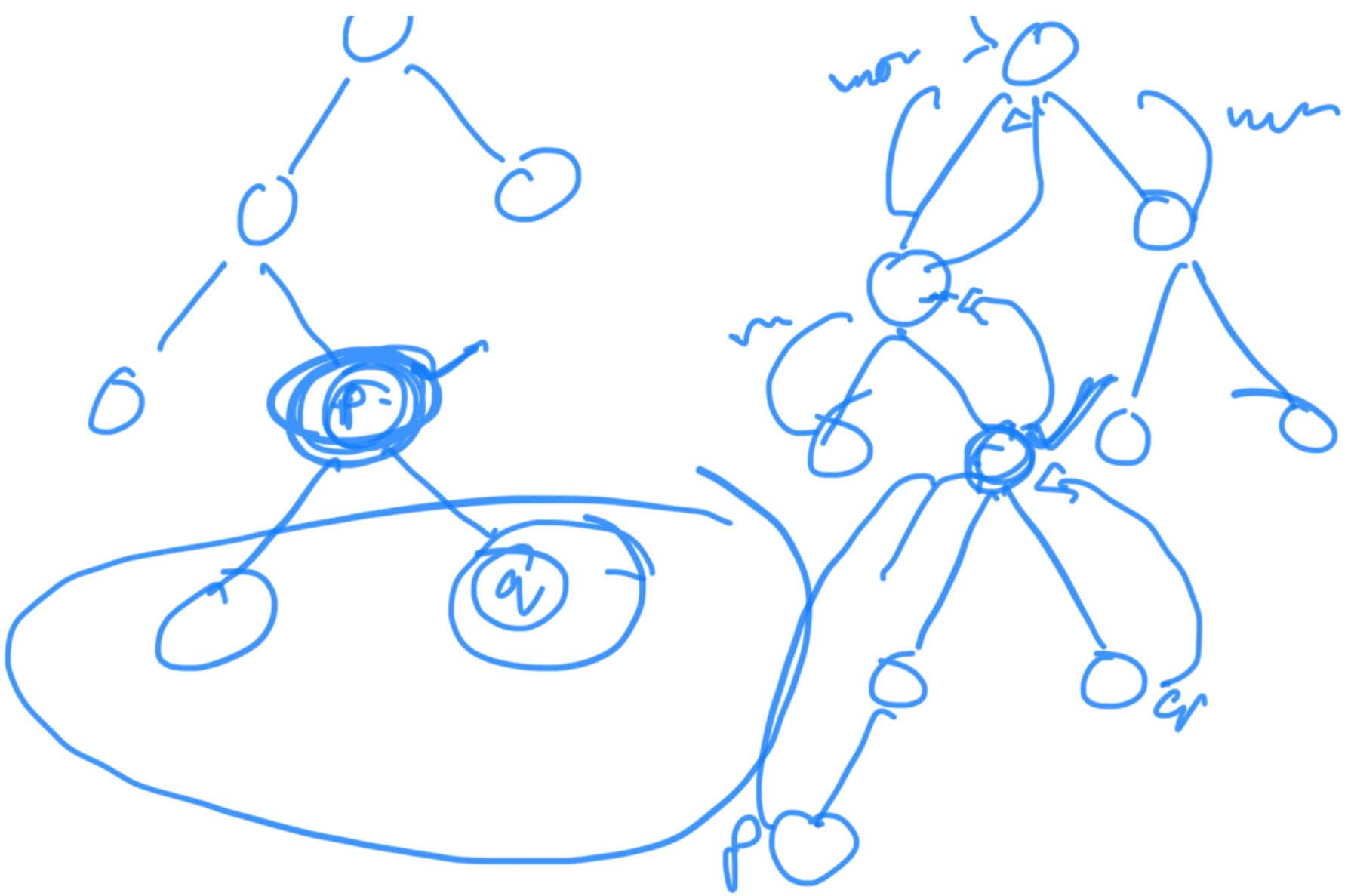


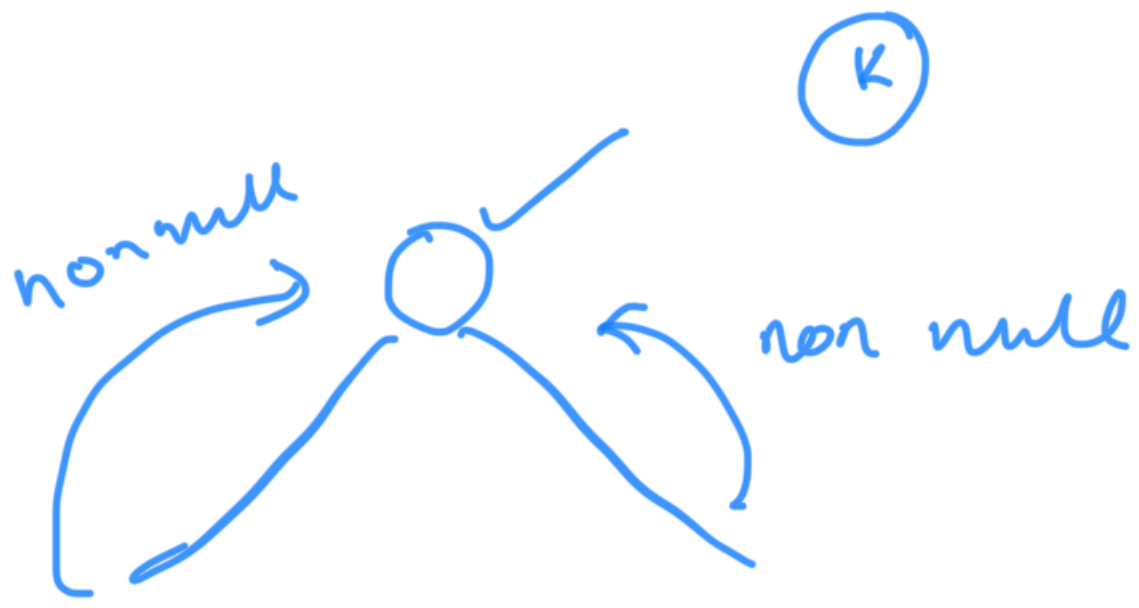
- post order traversal

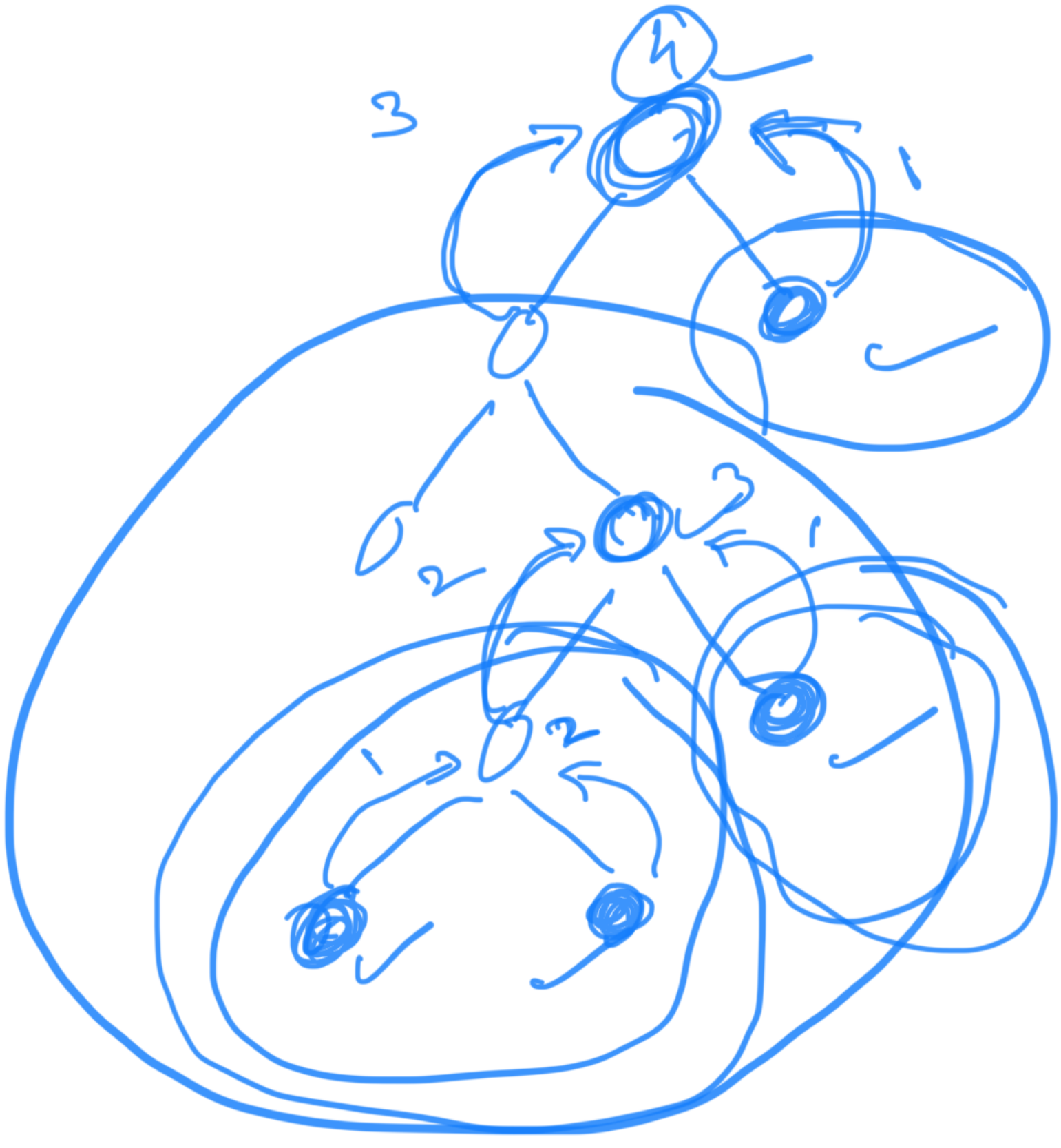
O(N)

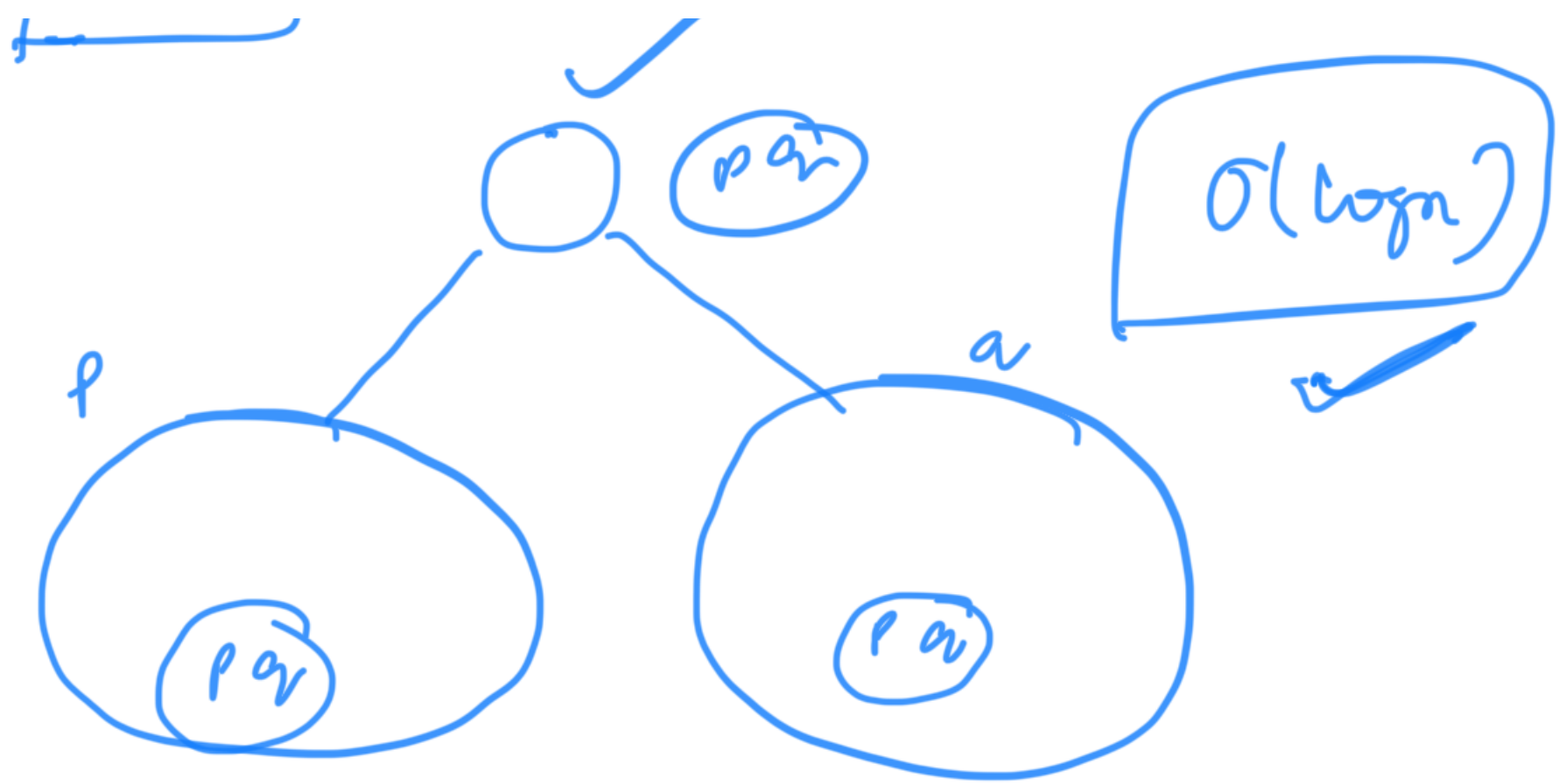
```
Node findLCA ( Node root, Node p, Node q )
{
    if ( root == null || p == null || q == null)
```

```
        return
    }

    if ( root == p || root == q ) {
        return root;
    }

    Node leftLCA = findLCA ( root.left, p, q );
    Node RightLCA = findLCA ( root.right, p, q );

    if( leftLCA != null && rightLCA != null)
        return root;

    return leftLCA != null ? leftLCA : rightLCA;

}
```

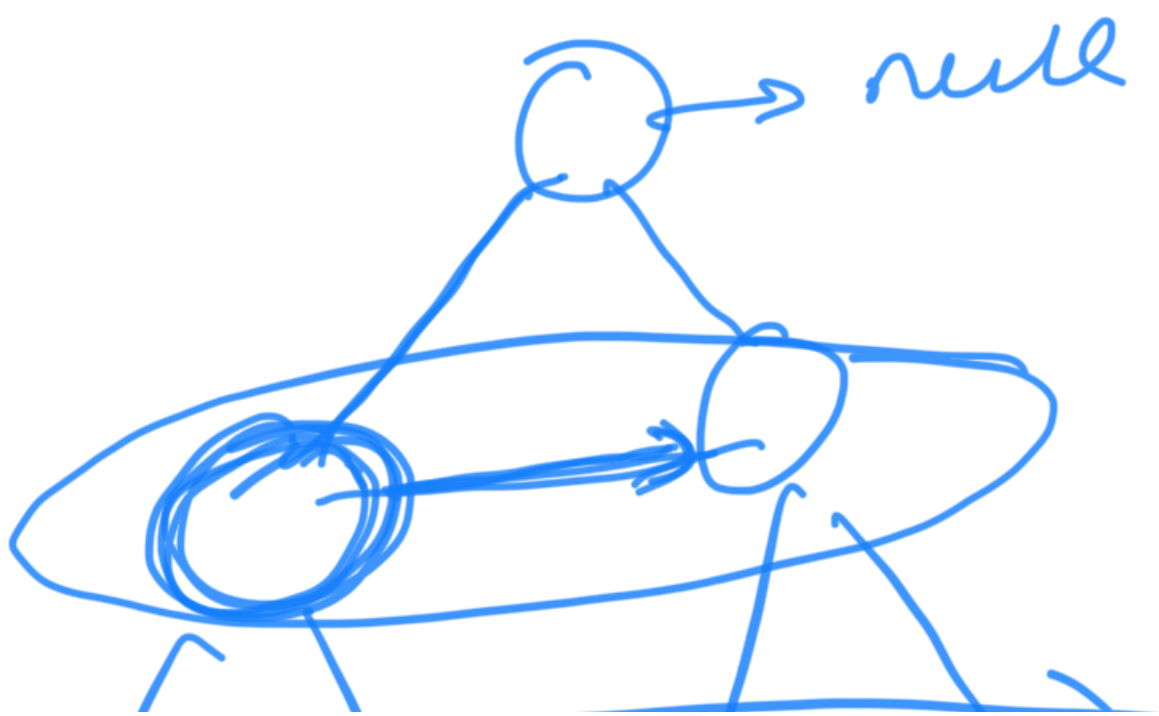

p
q
p

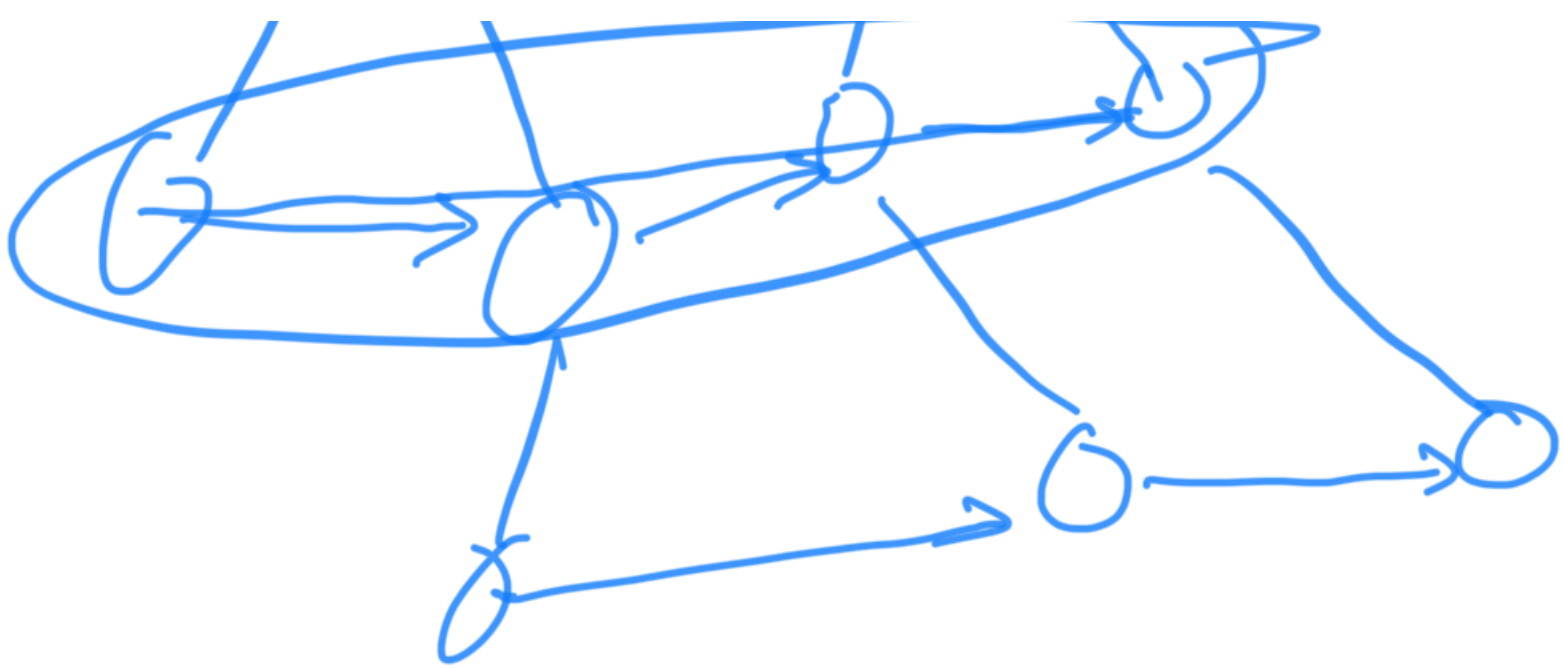

K
non null
non null

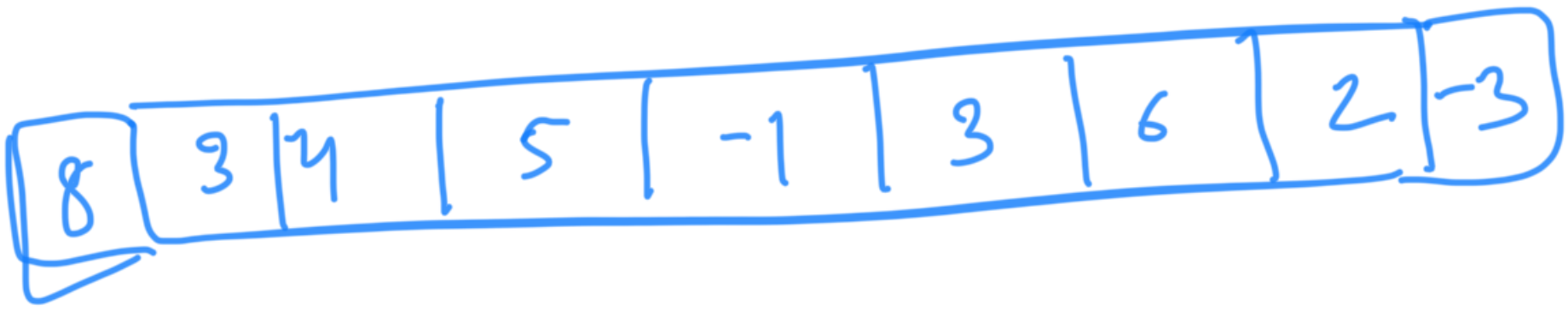

4
3
1
3
2
1
1
2
BST

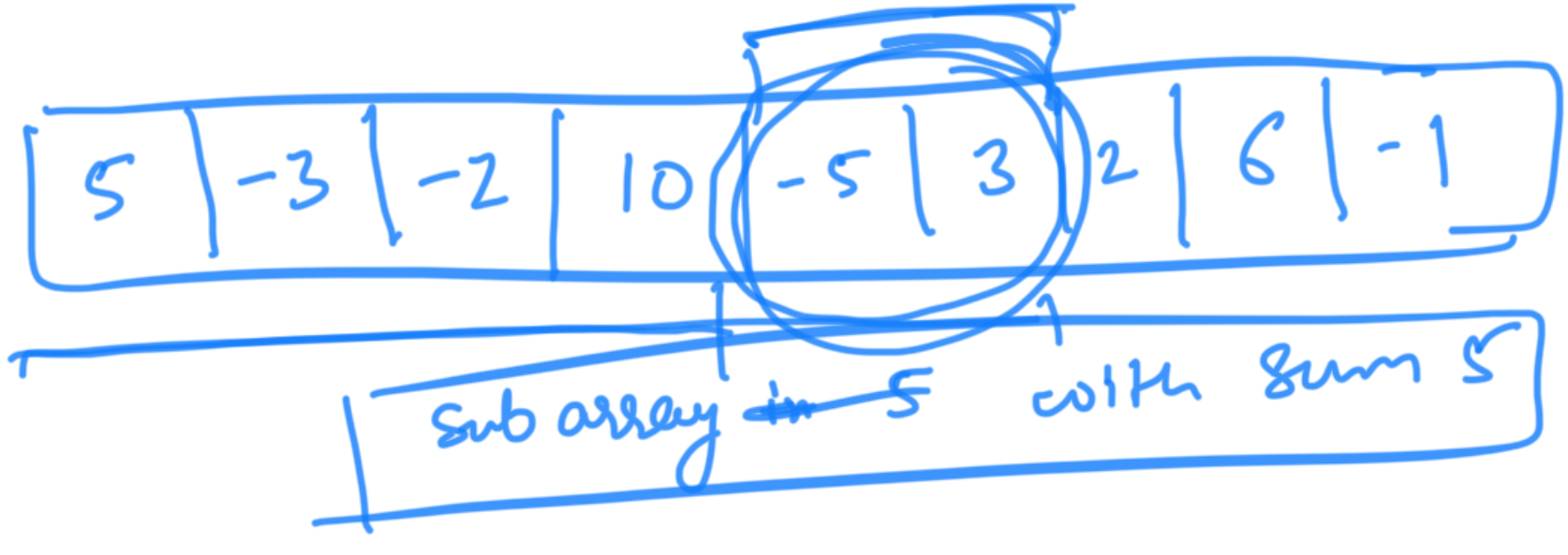


Populating next pointer









5 |

2
20
10
5
8
10
16
15

K = 3

Next class

Bit manipulation

~~segment tree~~

~~fenwick tree~~

Binary representation of Integer

Dynamic programming

n!=0 && (n & (n-1)) == 0

0 0 0 1 0 0 0 0

16 → 15

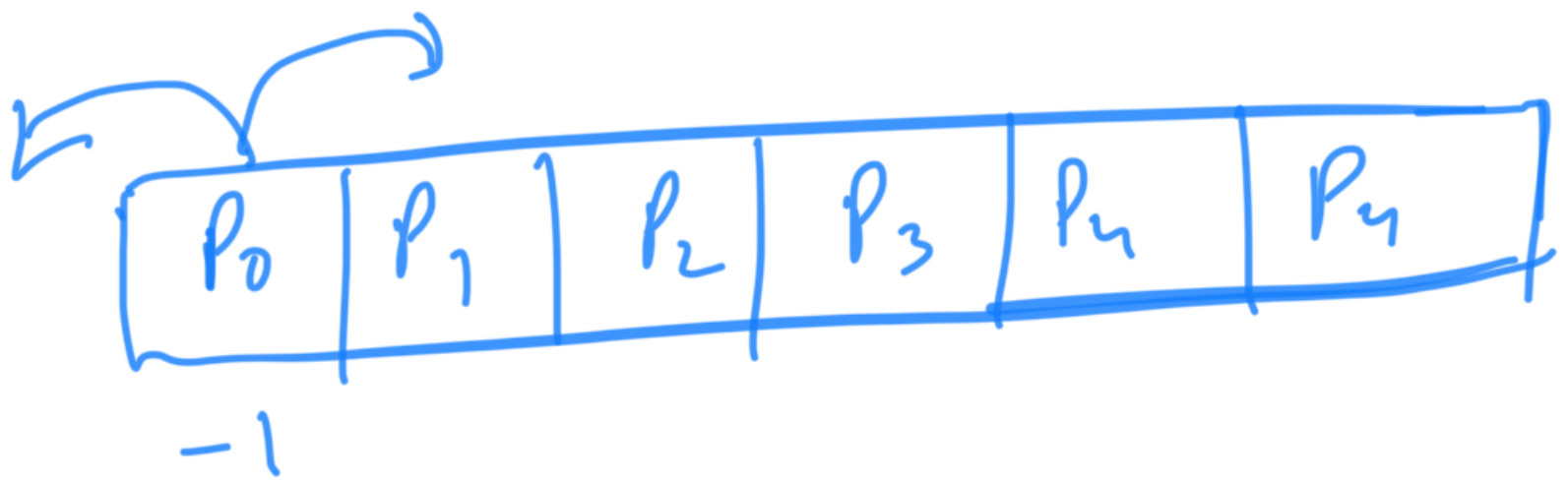
32 → 31

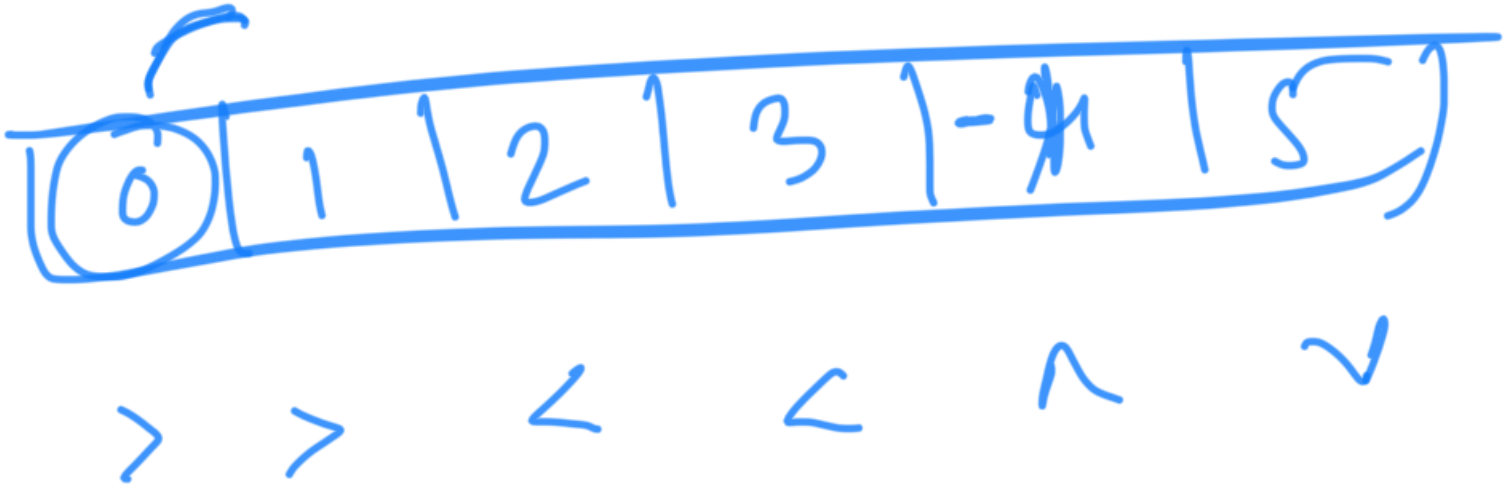
64 → 63

8

Map <0, 0>

0, 0
1, 1
2, 2


P0
P1
P2
P3
P4
P4
-1


0
1
2
3
-1
5
> > < < ^ v