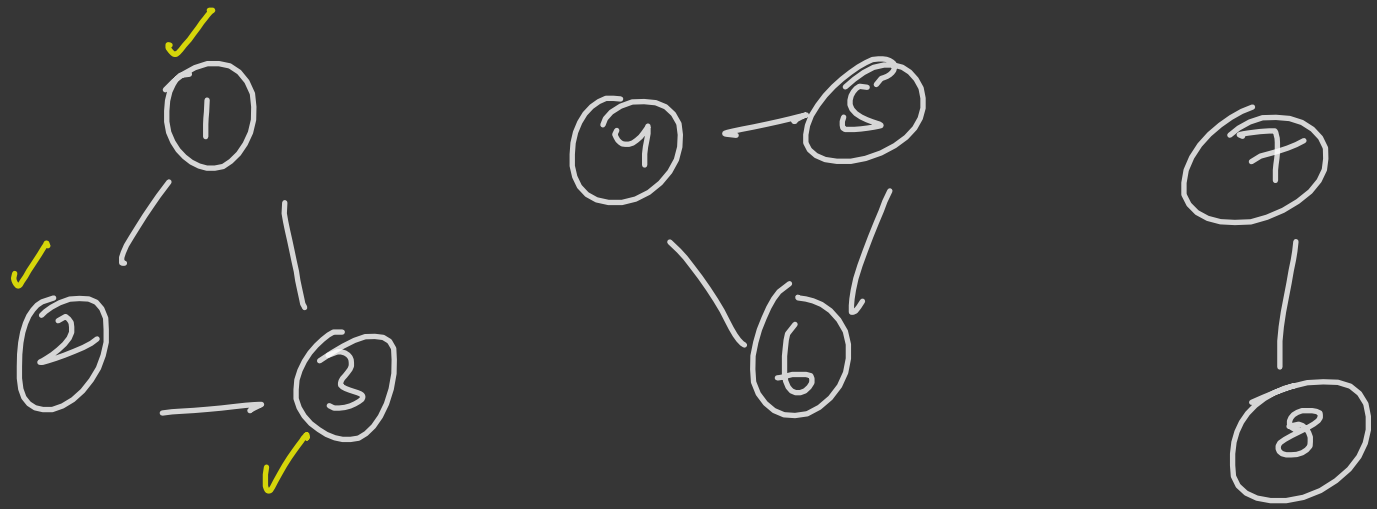
Disjoint Set Union

Dev Karan Singh (devkaran1231)
Expert at codeforces (1817)
5 star at codechef (2049)

Disjoint Set Union

- When is DSU required?
- How to find parent? (Path Compression technique)
- Union by size
- Union by rank
- Code Template of DSU
- Codeforces Problem
- SPOJ problem

Eg:



a b → a and b in same component or not?

1 2 → Yes

1 4 → No

single query → O(N)

↳ dfs from a/b

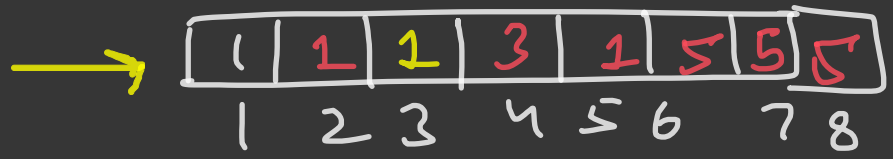
multiple queries

↳ N x q → TLE

↓

DSU

Eg: $n = 8$



(1, 2) union

(3, 4) union

(2, 4) union

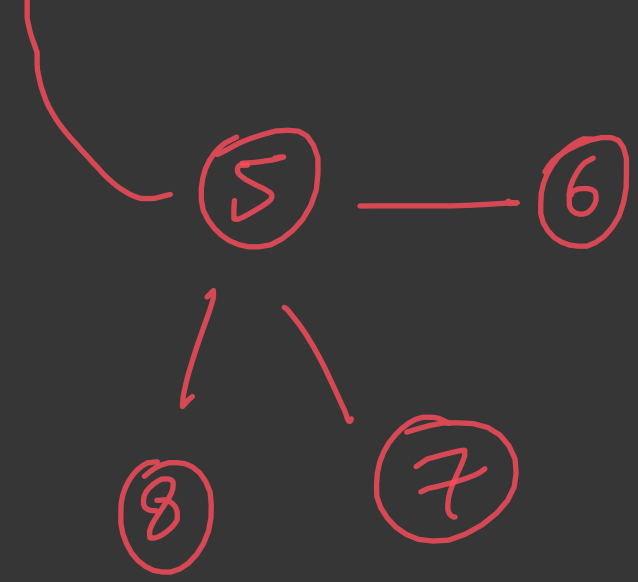
(5, 6) union

(7, 6) union

(6, 8) union

(8, 4) union

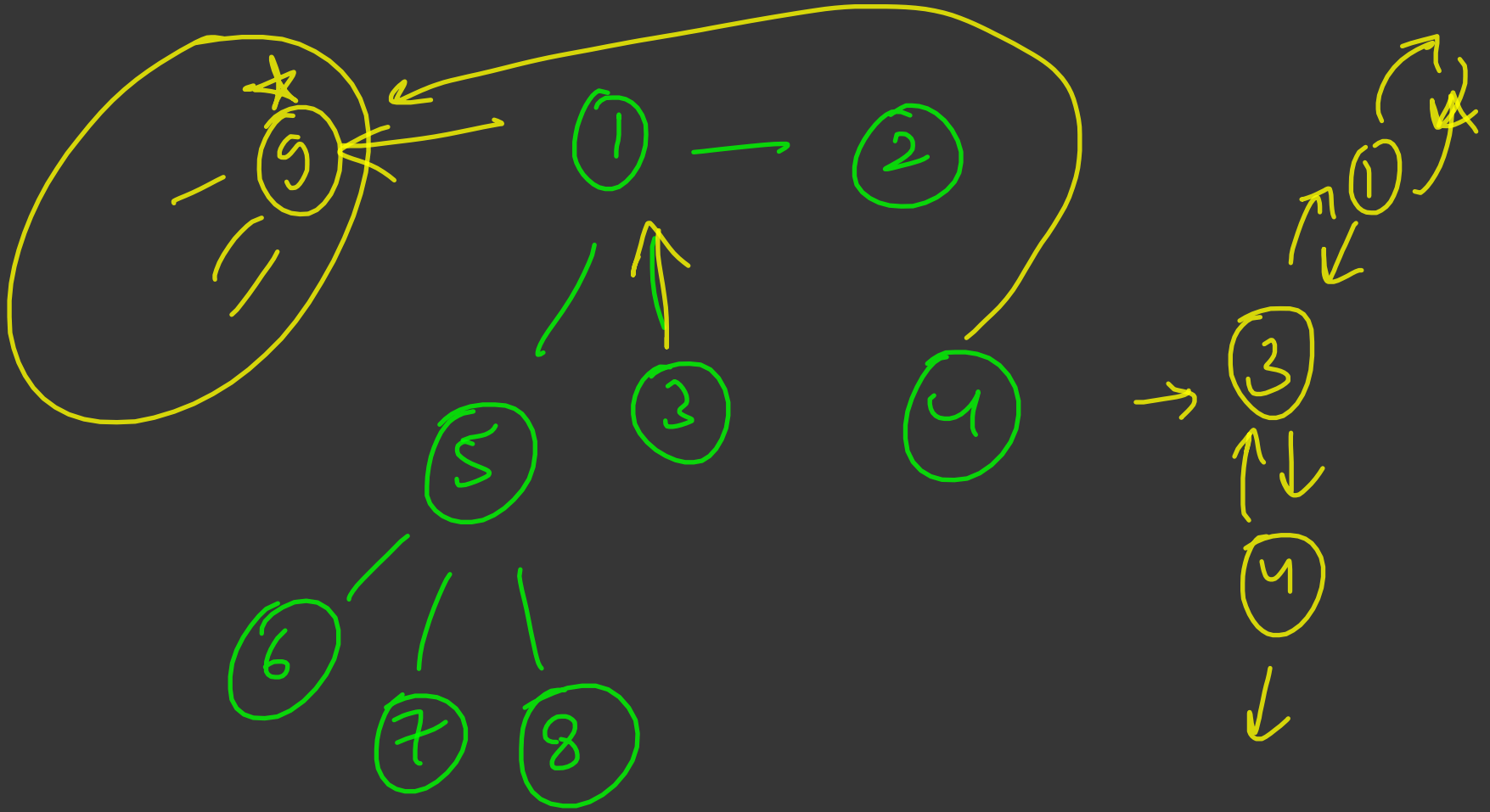
5 1

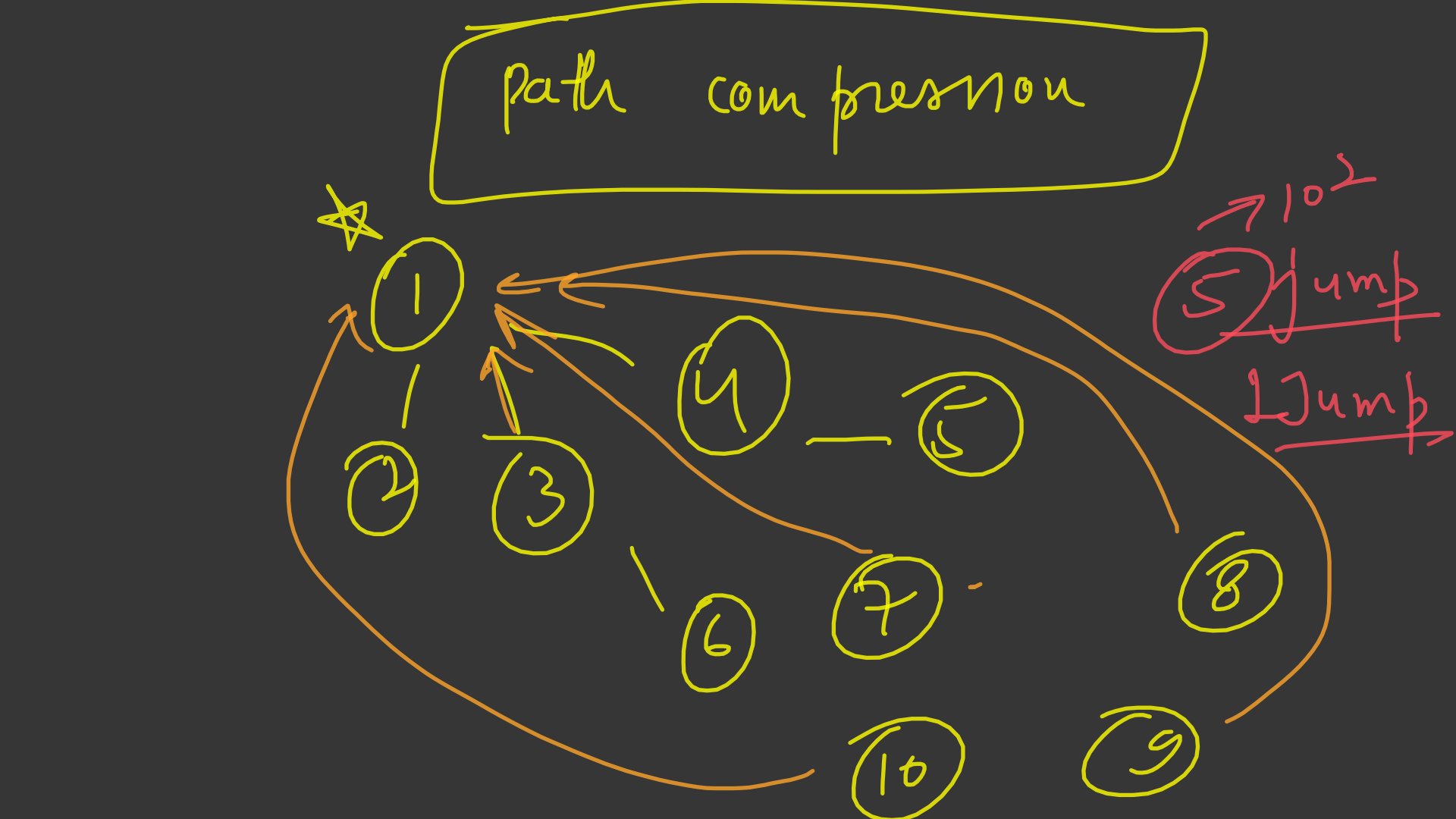


(6, 4) union

↳ same component

```
int findpar (int node){
    if (par[node] == node){
        return node;
    }
    return par[node] = findpar(par[node]);
}
```


9
1
2
3
4
5
6
7
8
1
3
4


path compression
1
2
3
4
5
6
7
8
9
10
10^2
5 Jump
1 Jump

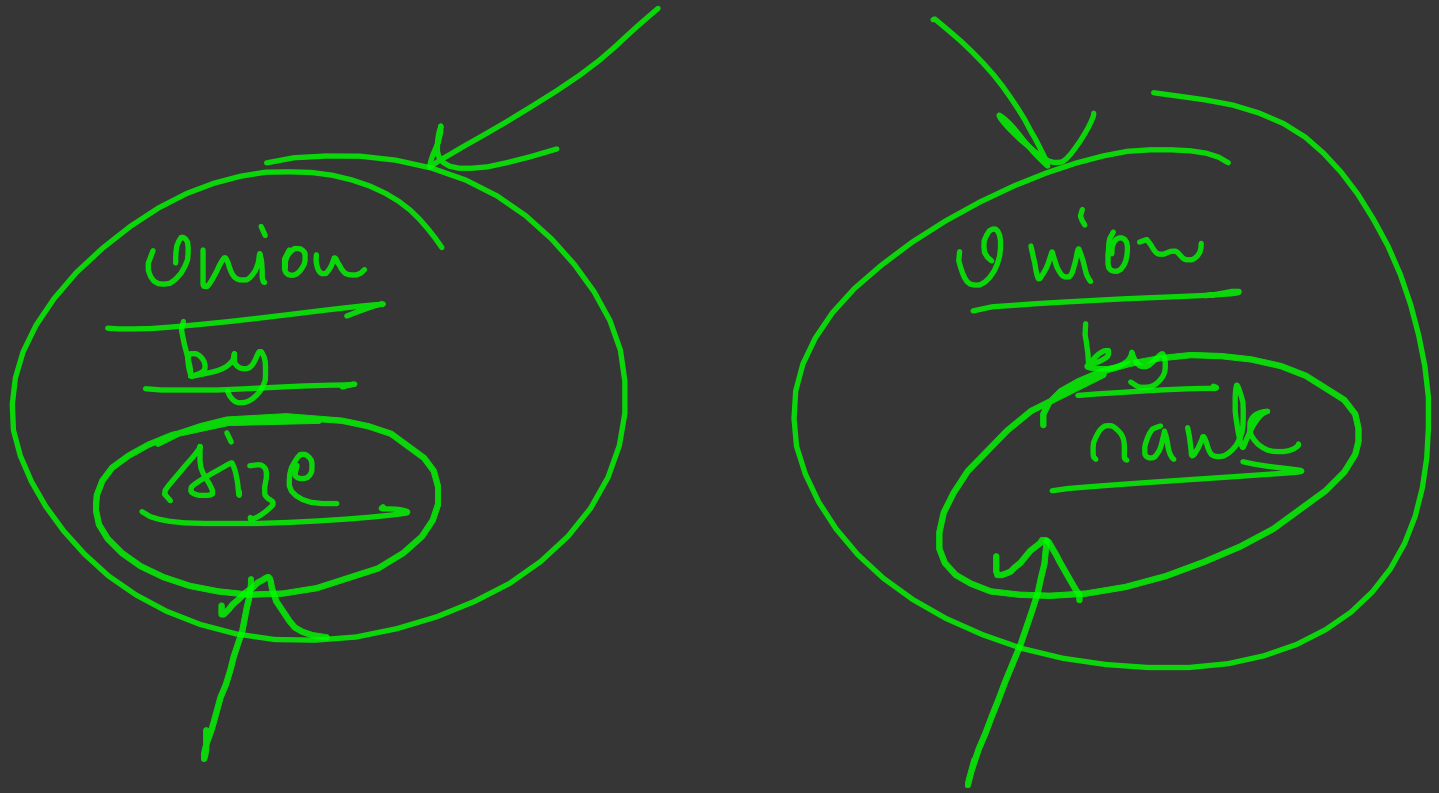
DP kind of approach

$O(N)$ —path comp.→ $O(4 \times)$

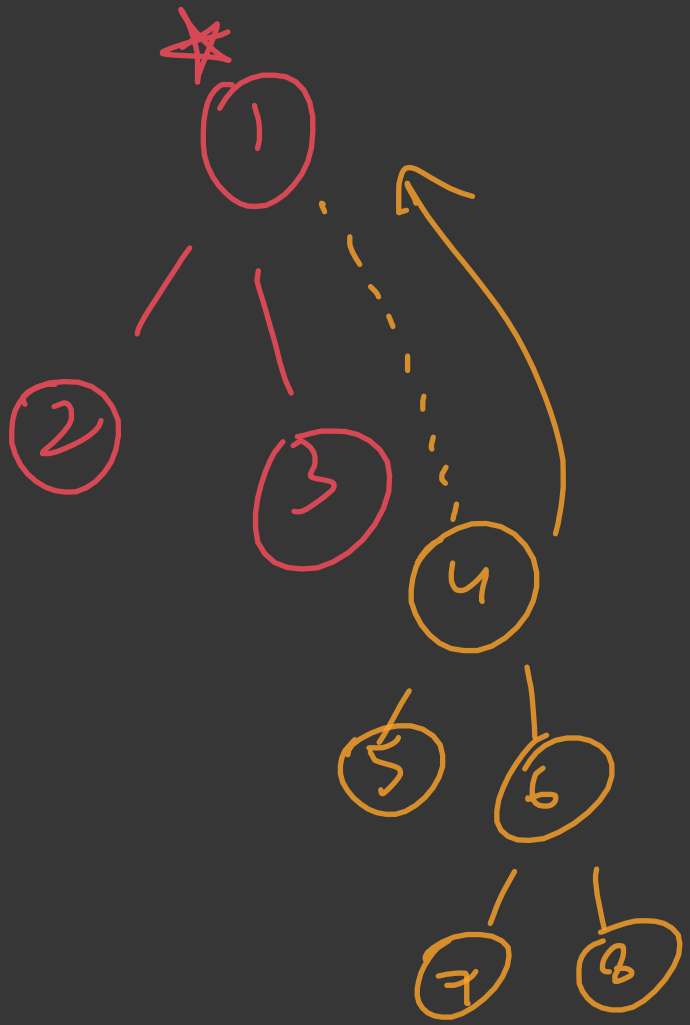
$O(4 \times)$ ← constant

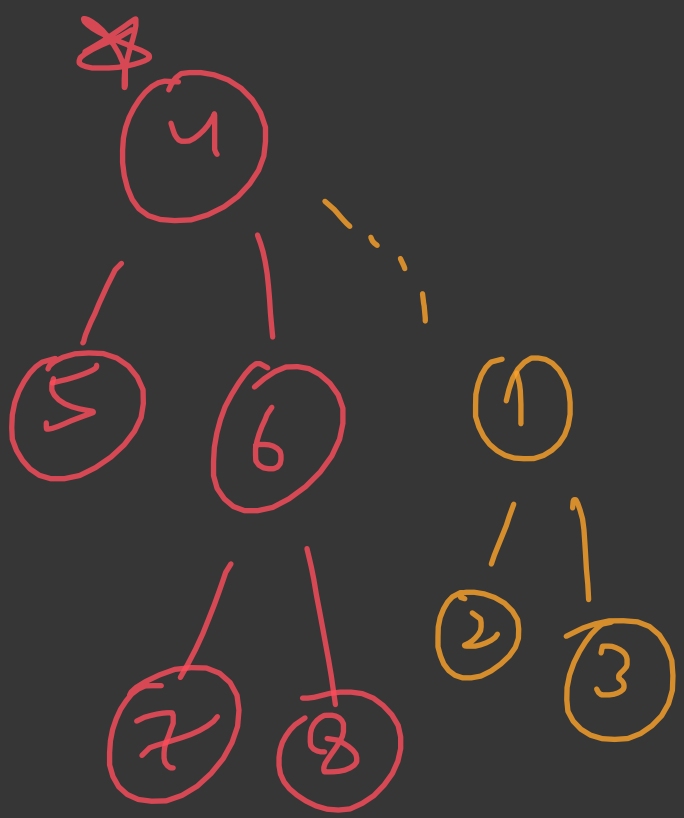
$\approx O(1)$

UNION METHODS



eg:





join smaller → bigger → optimal (greedy)

Q what is size?

size[i] → no. of nodes in i'th component


1
2
3
3
4
5
5
6
7
8
1 2 3 4 5 ... n

size

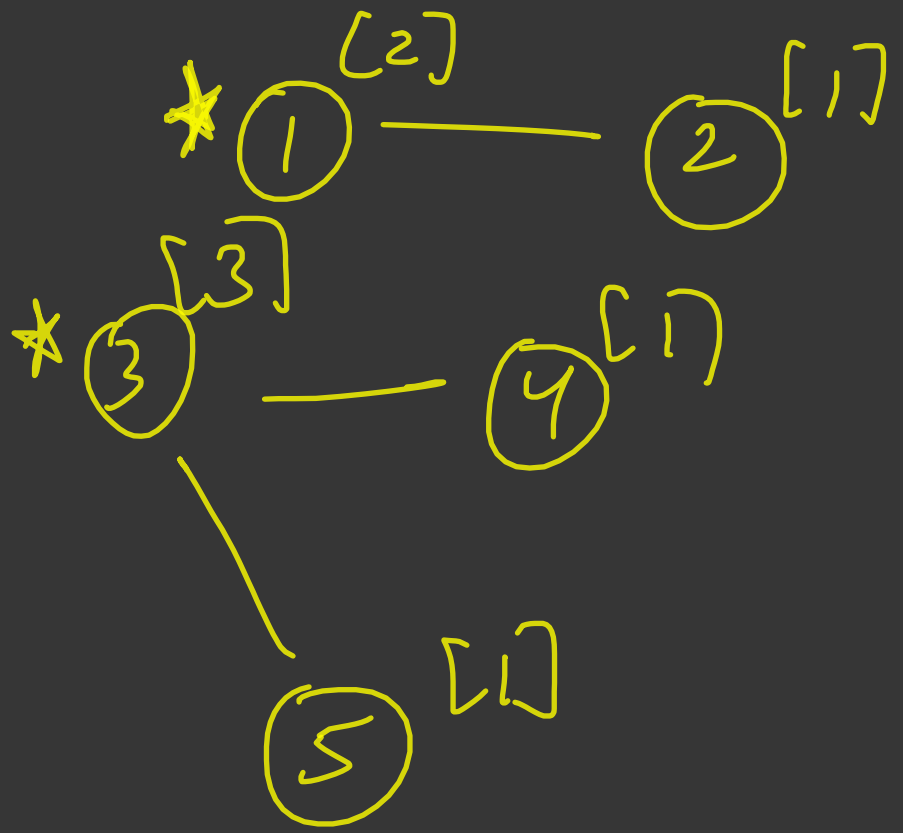
union(1, 2)

union(3, 4)

union(4, 5)

↓ ↓

3 5



rank → 0 for everyone.

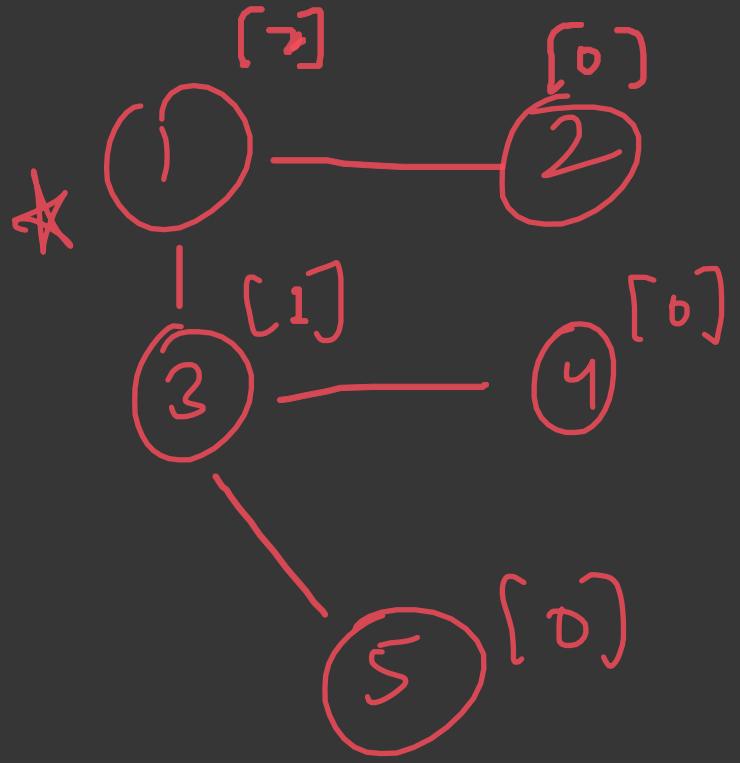
union(1, 2)

union(3, 4)

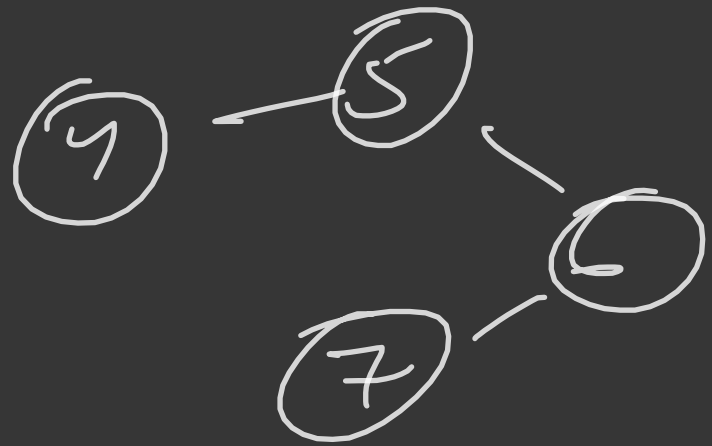
union(4, 5)

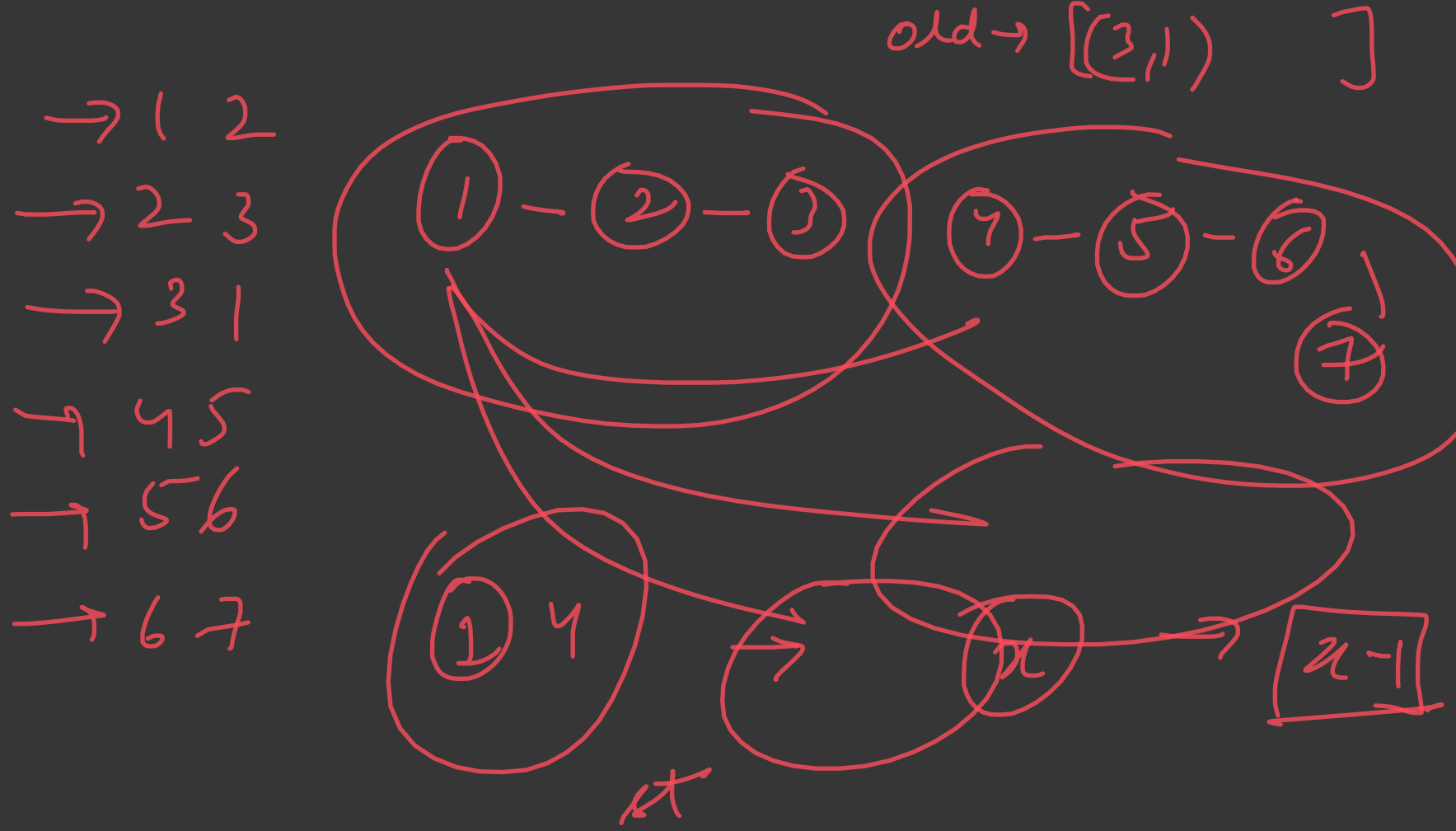
3 5

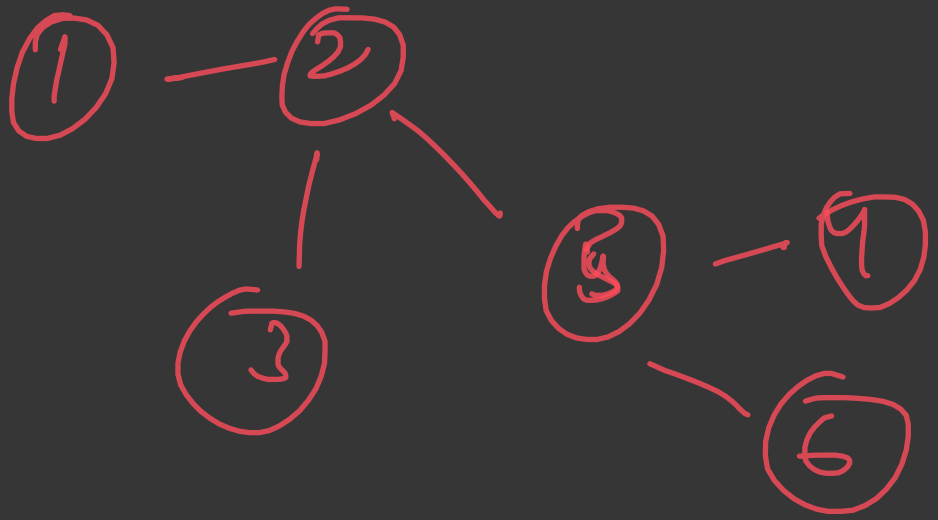
union(1, 3)

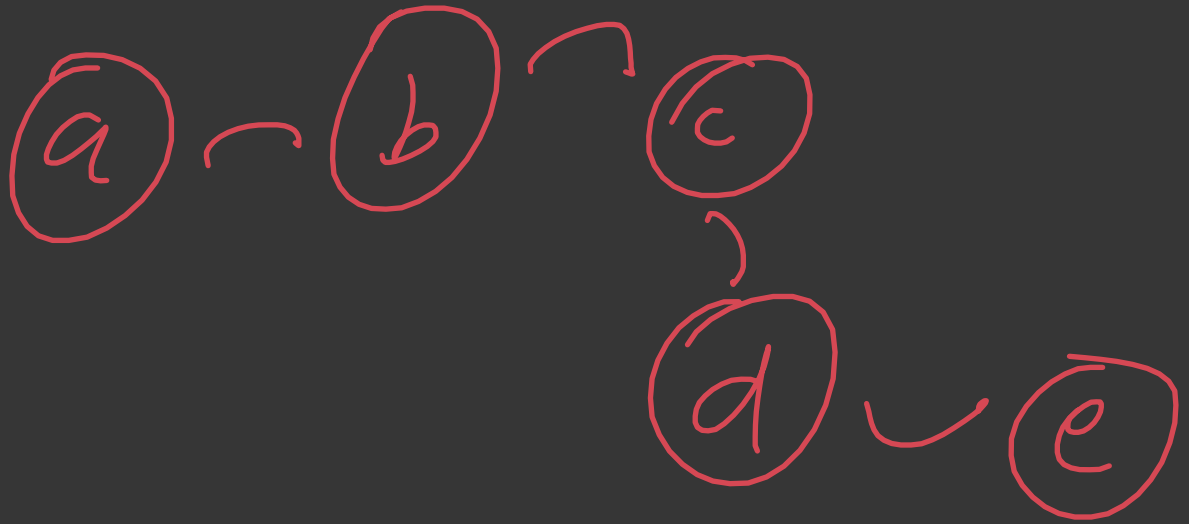


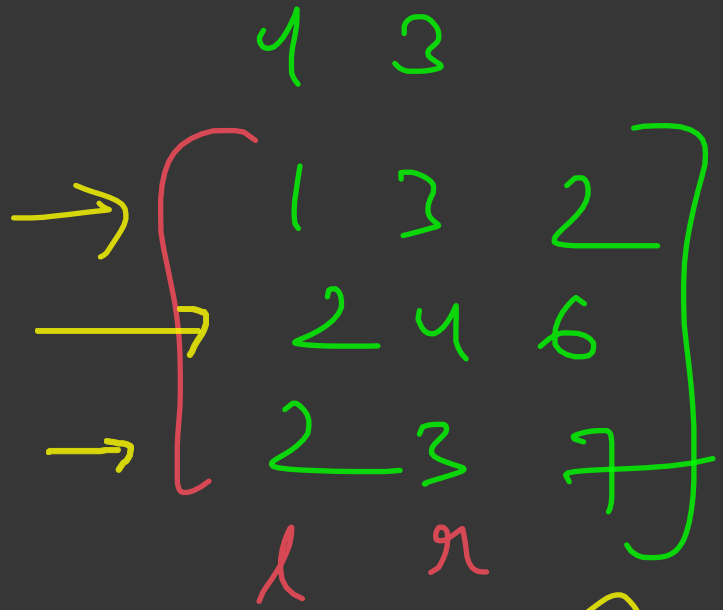

1
2
3

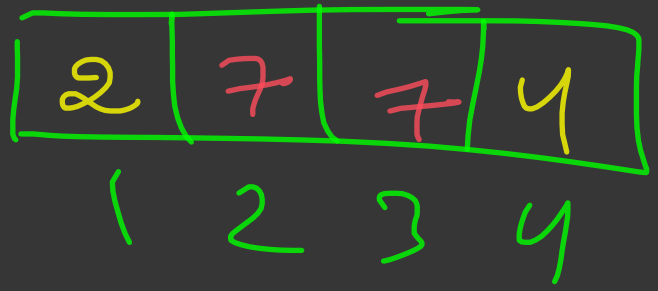

4
5
6
7

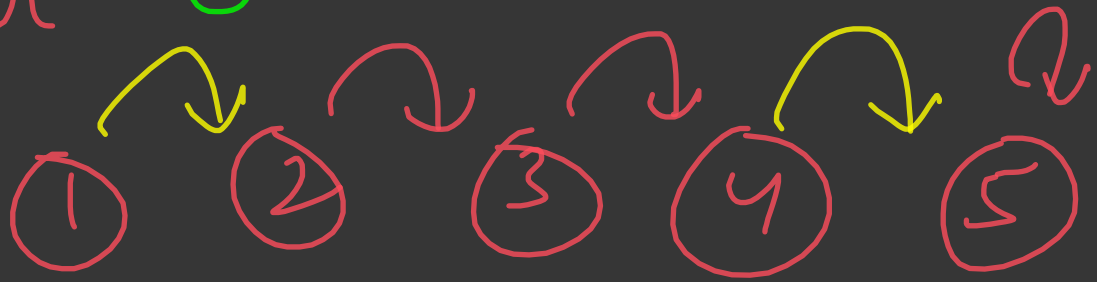

old → [(3,1)]
→ 1 2
→ 2 3
→ 3 1
→ 4 5
→ 5 6
→ 6 7
1
2
3
4
5
6
7
1
4
n
n-1


1
2
3
5
4
6


a
b
c
d
e


4 3
1 3 2
2 4 6
2 3 7
1 n


2 7 7 4
1 2 3 4


1 2 3 4 5