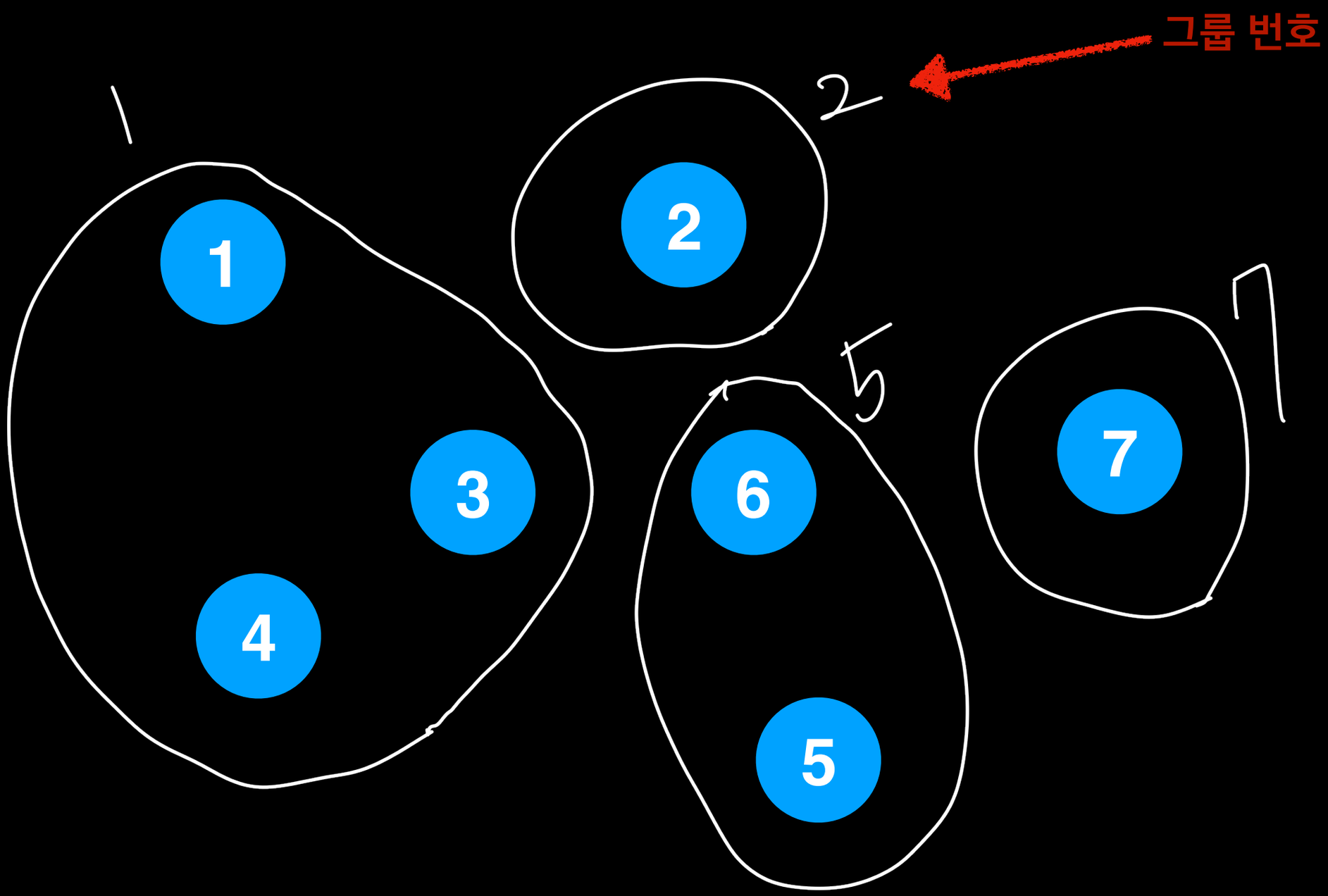
Union & Find

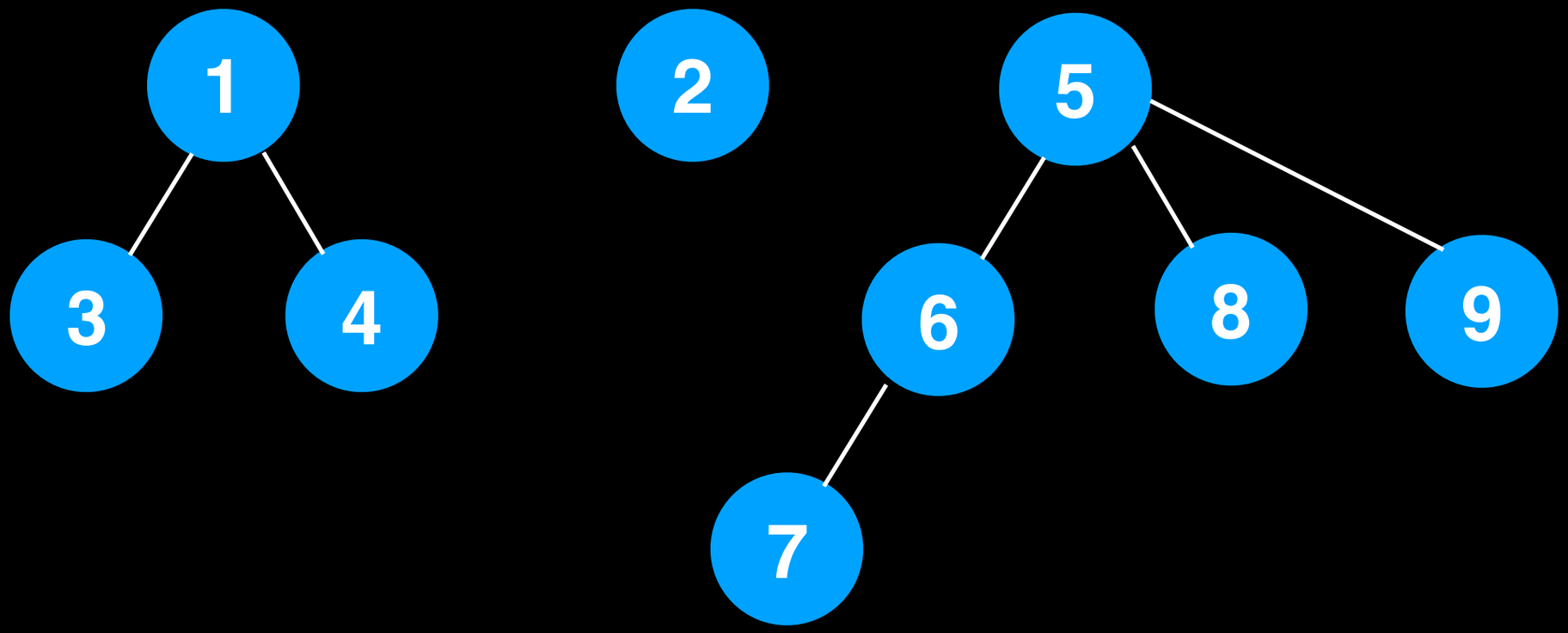
서로소 집합



역할

- find(x): x의 그룹 번호 찾기
- union(x, y): 원소 x와 y가 속한 그룹을 하나로 묶는다

구현



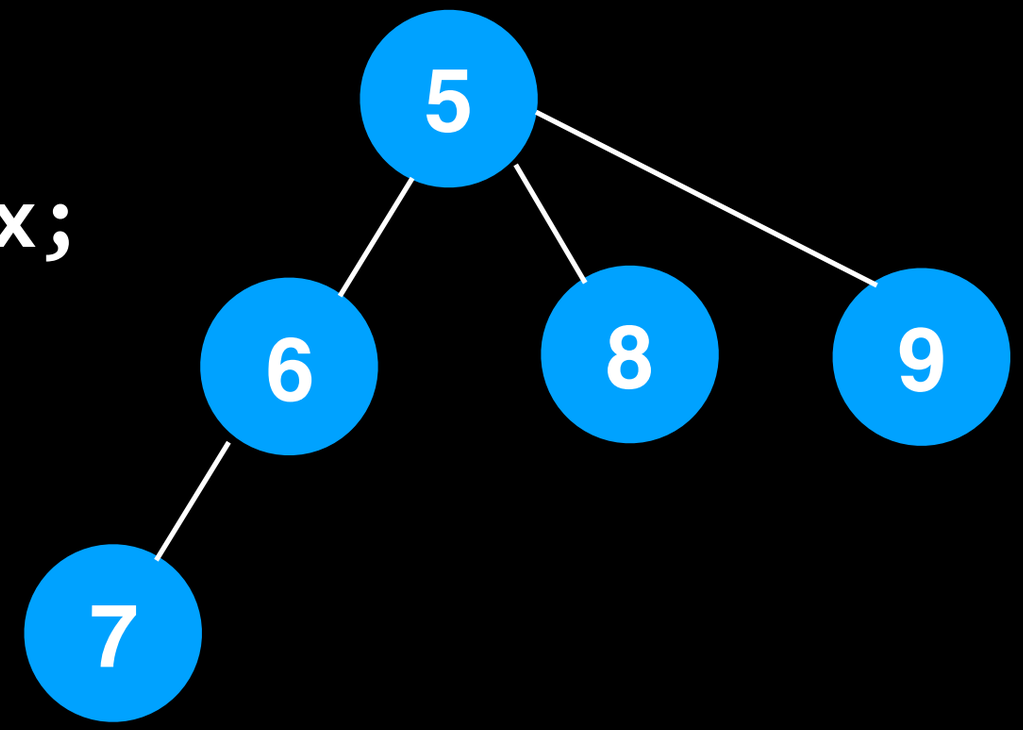
	1	2	3	4	5	6	7	8	9
P	1	2	1	1	5	5	6	5	5

부모를 저장하는 배열. 단, 루트는 자기 자신 저장

Find

- 부모를 계속 따라가면 된다.

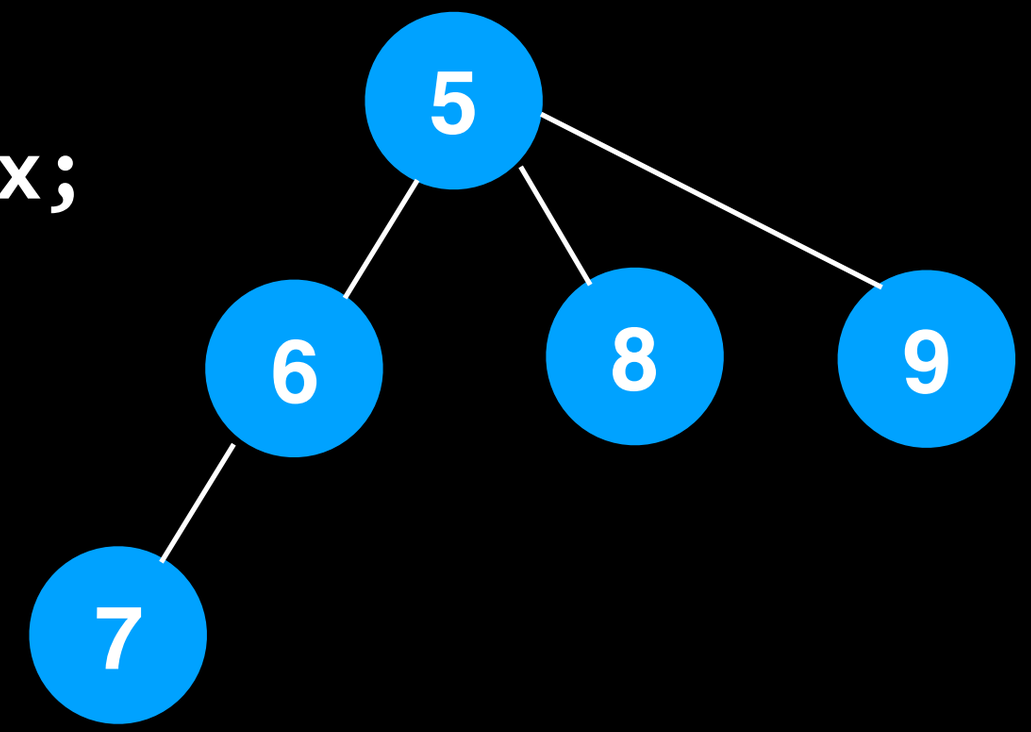
```
int Find(x){
    if(P[x] == x) return x;
    return Find(P[x]);
}
```



Find - 빠르게

- 부모를 계속 따라가면 된다.

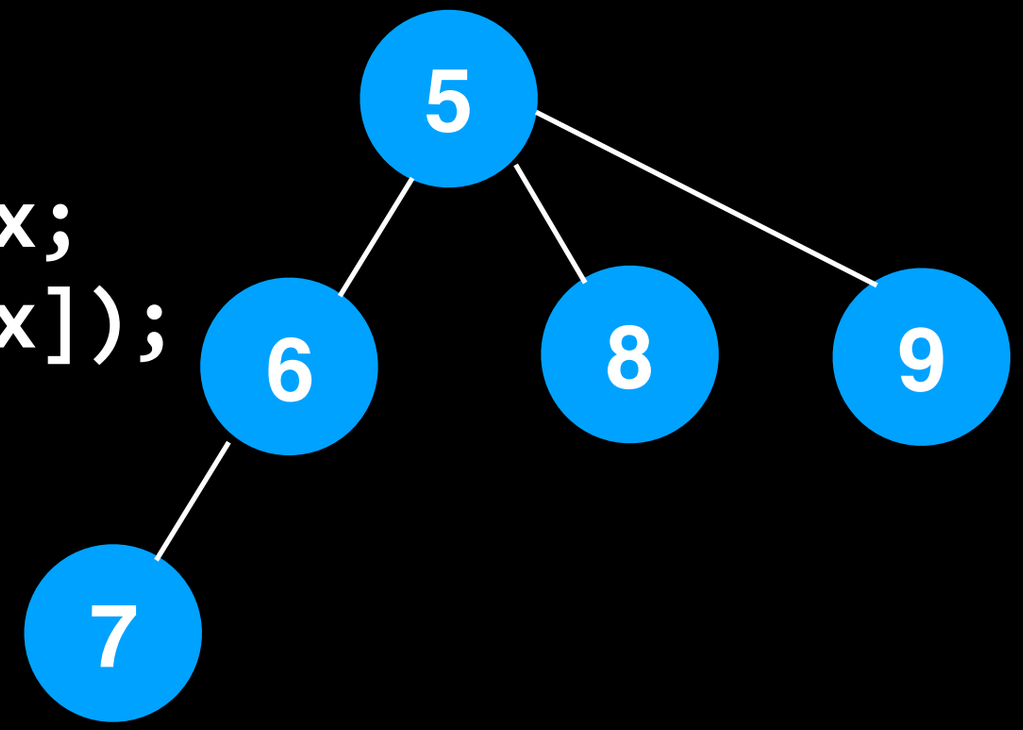
```
int Find(x){
    if(P[x] == x) return x;
    P[x] = Find(P[x]);
    return P[x];
}
```



Find - 빠르게

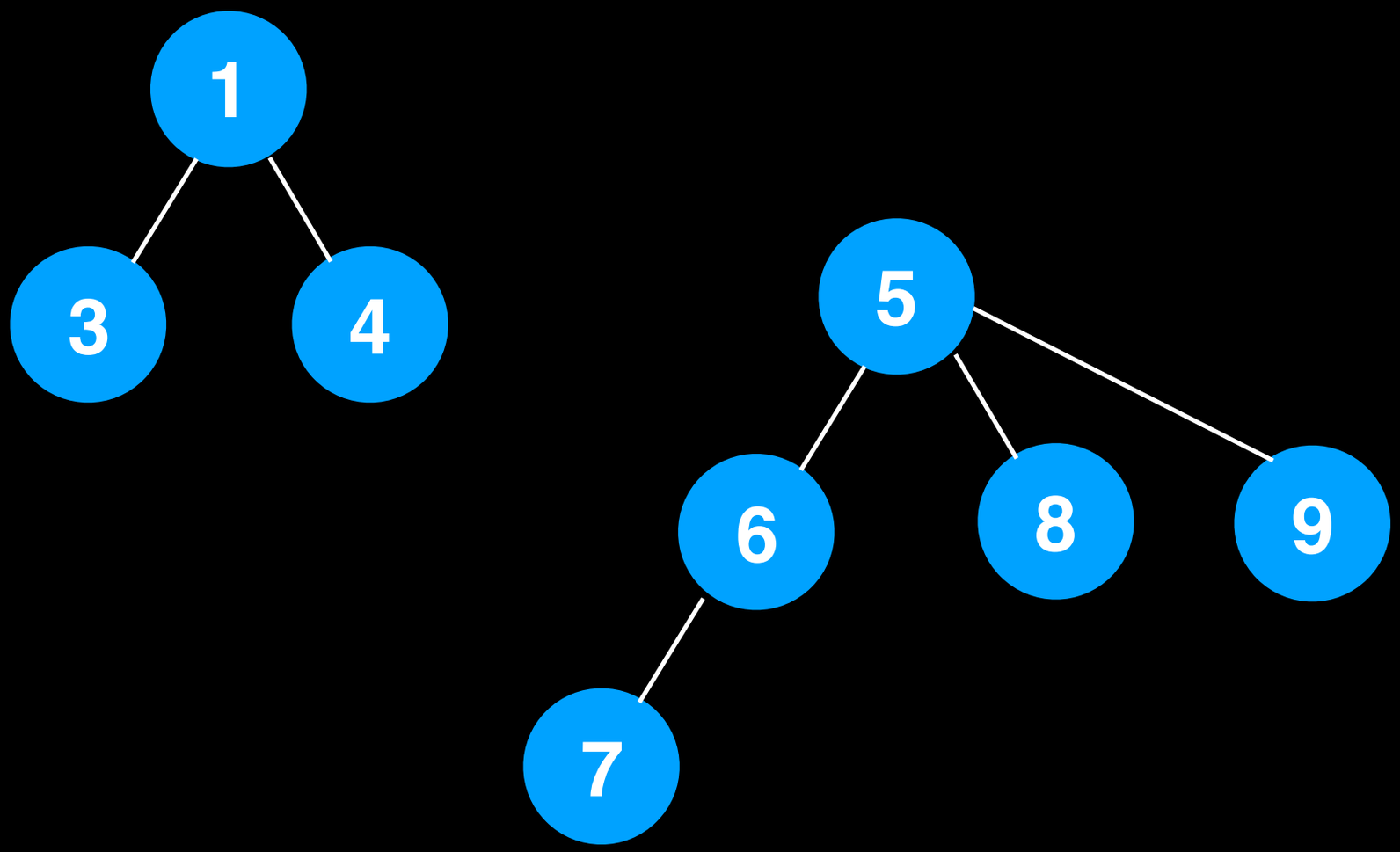
- 부모를 계속 따라가면 된다.

```
int Find(x){
    if(P[x] == x) return x;
    return P[x] = Find(P[x]);
}
```



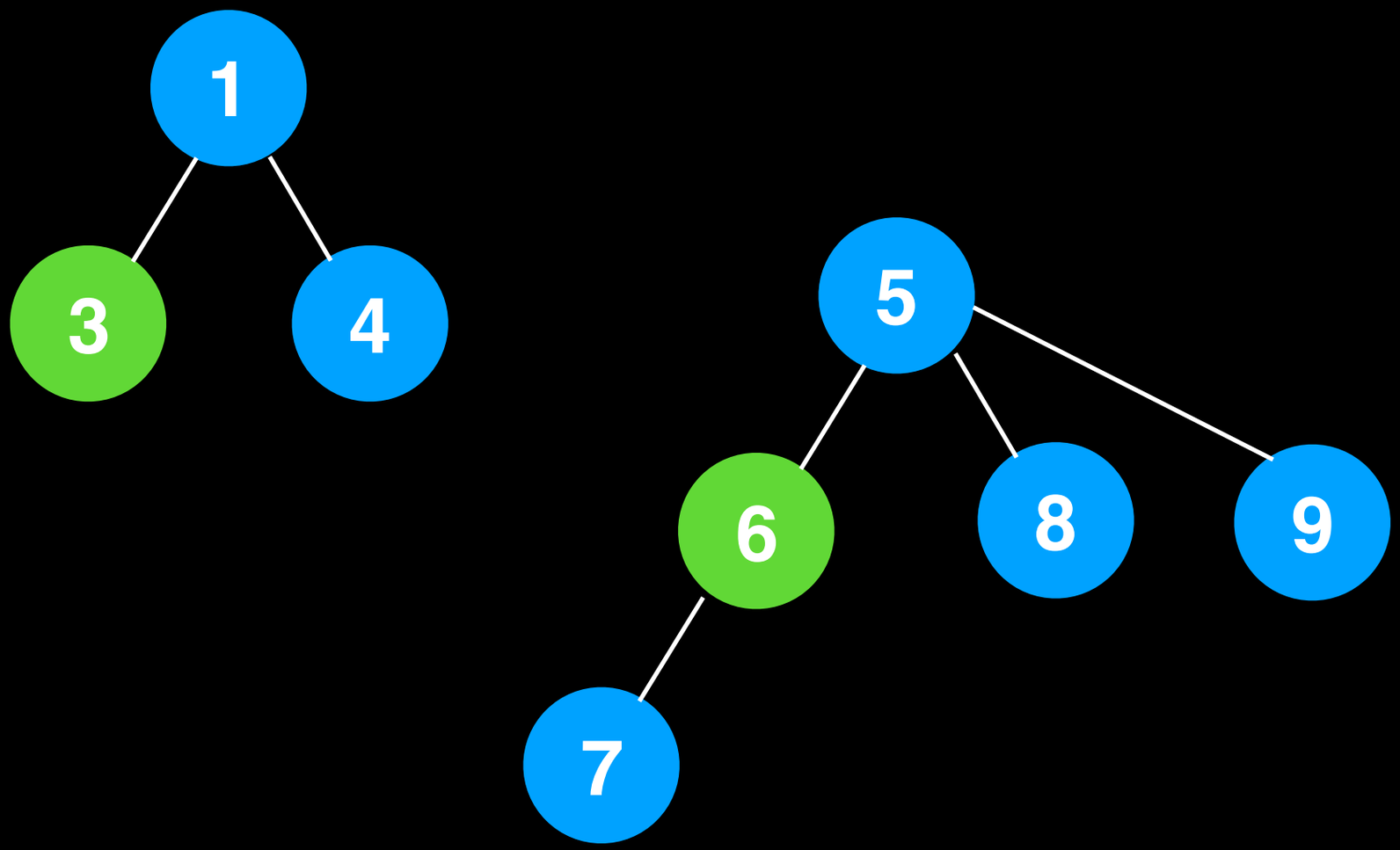
Union

```
Union(3, 6)
```



Union

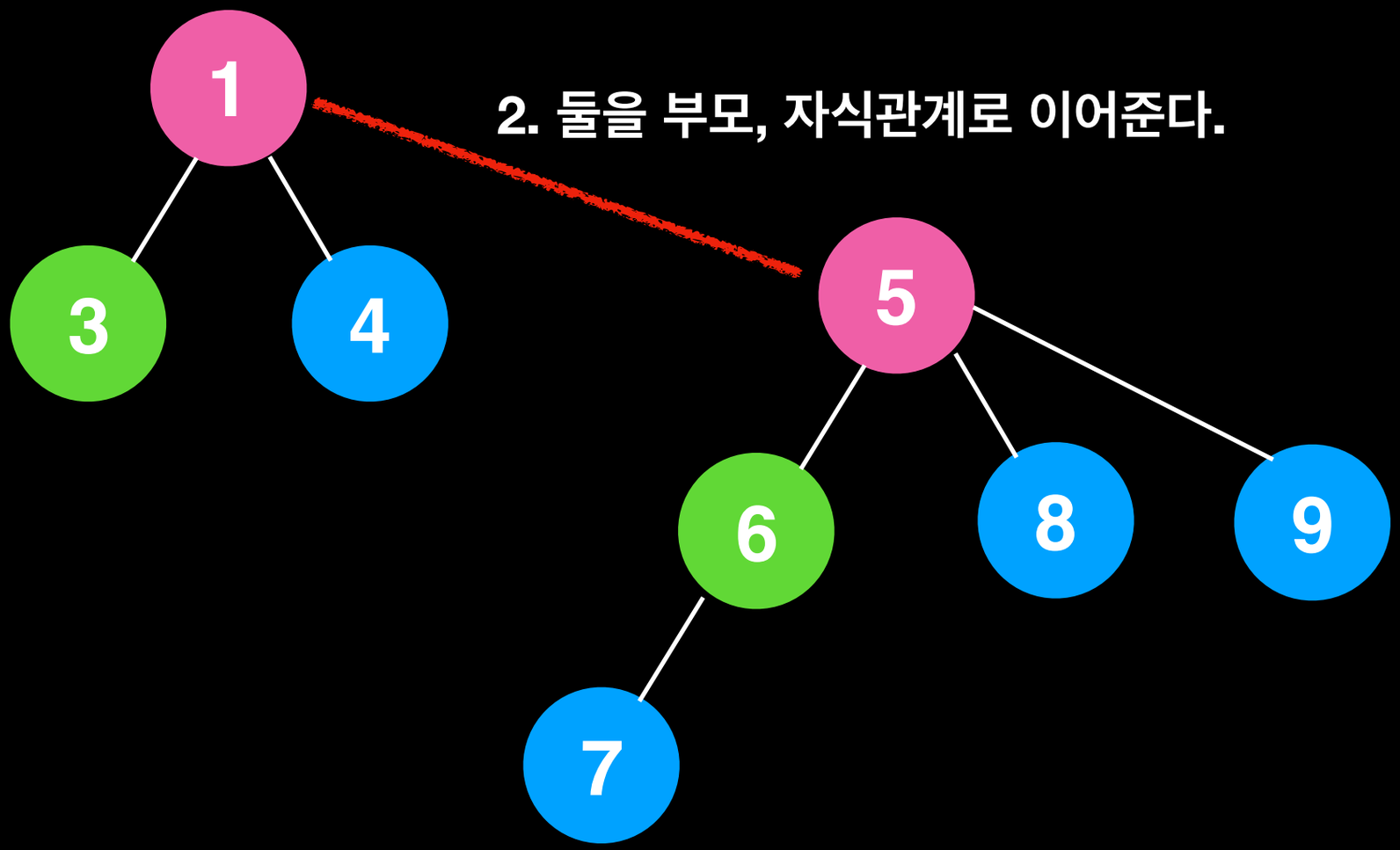
```
Union(3, 6)
```



Union

1. 최고부모를 찾는다.

Union(3, 6)



Union

```
void Union(int x, int y){
    int X = Find(x);
    int Y = Find(y);
    if(X != Y) P[X] = Y;
}
```