

```
struct DisjointSet {
    vector<int> parent, rank;
public://공개 멤버, 외부에서도 확인 가능
    DisjointSet(int n) : parent(n), rank(n, 1) {
        // 1. 초기화
        for (int i = 0; i < n; i++) {
            parent[i] = i;
        }
    }
    // find: u 가 속한 트리의 루트 노드 번호를 반환한다.
    int find (int u) {
        if (u == parent[u]) return u;
        return parent[u] = find(parent[u]);
    }
    // merge: u 가 속한 트리와 v 가 속한 트리를 합친다.
    void merge (int u, int v){
        u = find(u);
        v = find(v);
        // u와 v의 root가 같으면 merge X
        if (u == v) return;
        if (rank[u] > rank[v]) swap(u, v);
        parent[u] = v;
        // 두 트리의 높이가 같은 경우에는 결과 트리의 rank 를 1 높여준다.
        if (rank[u] == rank[v]) ++rank[v];
    }
};
```