```cpp
class Union_Find {
private:
  vi p, r, c; // parent, rank, the number of connected components
public:
  Union_Find(int N) {
    p.resize(N);
    r.resize(N);
    c.resize(N);
    Loop(i, N) {
      p[i] = i;
      r[i] = 0;
      c[i] = 1;
    }
  }
  int find(int x) {
    if (p[x] == x) return x;
    else return p[x] = find(p[x]);
  }
  void unite(int x, int y) {
    x = find(x);
    y = find(y);
    if (x == y) return;
    if (r[x] == r[y]) r[x]++;
    if (r[x] < r[y]) {
      p[x] = y;
      c[y] += c[x];
    }
    else {
      p[y] = x;
      c[x] += c[y];
    }
  }
  bool is_same(int x, int y) {
    return find(x) == find(y);
  }
  int get_cnt(int x) {
    return c[find(x)];
  }
};
```