7/10/20

Chandan A.M.
1BM18CS025

③ Number of Islands:

```
class Disjoint {
    int n, count ;
    vector <int> parent ;
    public Disjoint (int n) {
        parent [n] ;
        for( int i =0; i<n; i++) {
            parent [i] = i ;
        }
        count = 0;    }
    public int find (int x) {
        if (parent [x] != x ) {
            return find (parent [x]) ;
        }
        return x ;
    }

    public void join( int x, int y) {
        int Root x = find(x);
        int Rooty = find(y);
        if ( Root x != Rooty ) {
            parent [Root x] = Root y ;
            count -- ;
        }
    }
    public void counter (int n) {
        count = n ;
    }
    public int count () {
        return count;
    }
 };


int countIslands( vector< vector<int>> m){
    int count =0;
    int a = m.size();
    int b = m[0].size();
    for(int i=0; i<a; i++){
        for(int j=0; j<b; j++){
            if( a[i][j] == 1)
                // check all the neighbouring
                // values. If value is 1,
                // take the union of the
                // index and its neighbour
            }
        }
    }
    Disjoint d= new Disjoint (a,b);
    for (int i=0; i<a; i++){
        for(int j=0; j<b; j++) {
            if(m[i][j]){
                count++;
            }
        }
    }
    d.counter(count);
    return d.count();
}
```