```
import java.util.Arrays;
import java.util.HashMap;
import java.util.Map;

public class FriendCircle {
    private String[] findLargestFriendCircle(String[][] arr1) {

        Map<String, String> unionKeys = new HashMap<>();
        for(String []pair:arr1) {
            unionKeys.put(pair[0],pair[0]);
            unionKeys.put(pair[1],pair[1]);
        }

        for(String[] pair: arr1) {
            String unionKey = null, other = null;
            String key0 = unionKeys.get(pair[0]), key1 = unionKeys.get(pair[1]);
            if(key0.compareTo(key1) < 0) {
                unionKey = key0;
                other = pair[1];
            } else {
                unionKey = key1;
                other = pair[0];
            }
            unionKeys.put(other, unionKey);
        }

        Map<String, Integer> counts = new HashMap<>();
        for(Map.Entry<String, String> ent : unionKeys.entrySet()) {
            counts.put(ent.getValue(), counts.getOrDefault(ent.getValue(), 0) + 1);
        }

        int max = -1;
        String maxKey = null;
        for(Map.Entry<String, Integer> ent : counts.entrySet()) {
            if(ent.getValue() > max || (ent.getValue() == max && ent.getKey().compareTo(maxKey) <
0)) {
                max = ent.getValue();
                maxKey = ent.getKey();
            }
        }

        String[] res = new String[max];
        for(Map.Entry<String, String> ent : unionKeys.entrySet()) {
            if(ent.getValue().equals(maxKey)) {
                res[--max] = ent.getKey();
            }
        }

        Arrays.sort(res);
        return res;
    }
}
```