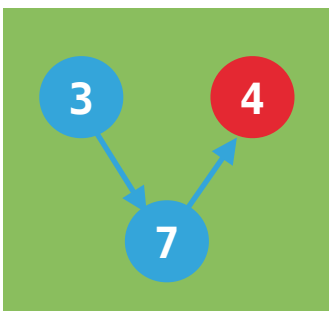
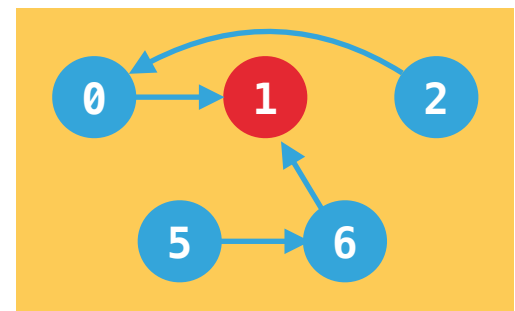


QUESTION 5.1.1 UNION-FIND: QUICK-UNION

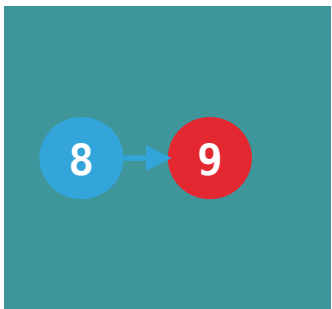
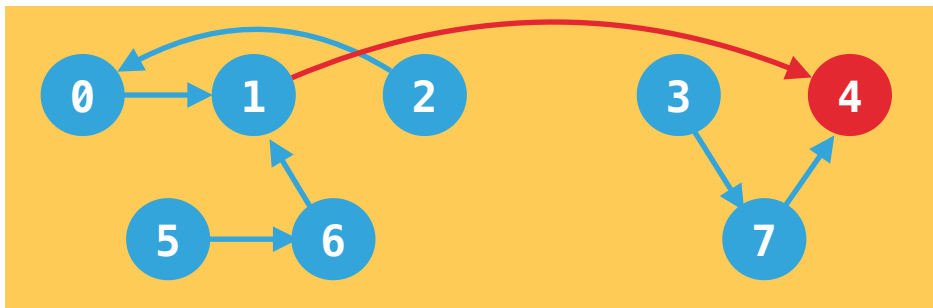
Val	0	1	2	3	4	5	6	7	8	9
-----	---	---	---	---	---	---	---	---	---	---

Rep 1 1 0 7 4 6 1 4 9 9



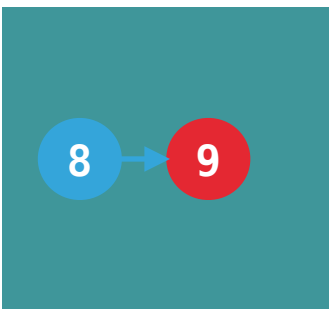
Val	0	1	2	3	4	5	6	7	8	9
-----	---	---	---	---	---	---	---	---	---	---

Rep 1 4 0 7 4 6 1 4 9 9



```
int find(int a) { //O(n)
    if(tab[a] != a)
        return find(tab[a]);
    return a;
}
```

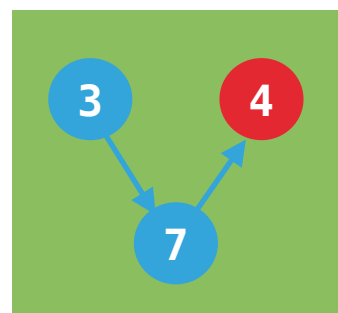
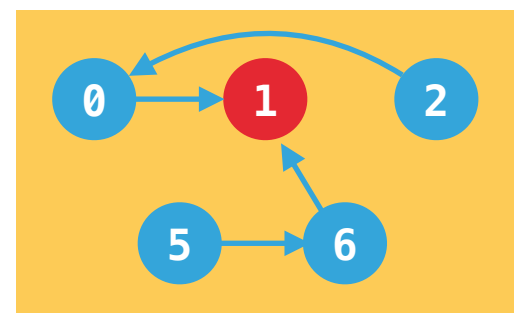
```
void union(int a, int b) { //O(1) ?
    int i = find(a);
    int j = find(b);
    tab[i] = j;
}
```



QUESTION 5.1.1 UNION-FIND: WEIGHTED QUICK-UNION

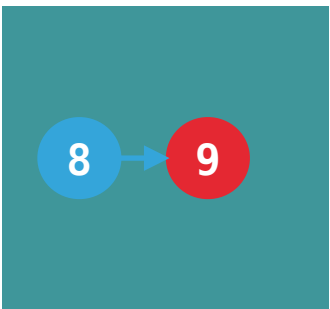
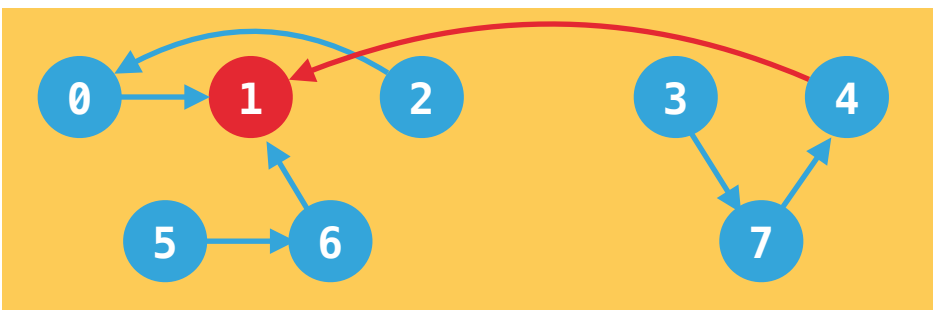
Val	0	1	2	3	4	5	6	7	8	9
-----	---	---	---	---	---	---	---	---	---	---

Rep 1 1 0 7 4 6 1 4 9 9



Val	0	1	2	3	4	5	6	7	8	9
-----	---	---	---	---	---	---	---	---	---	---

Rep 1 1 0 7 1 6 1 4 9 9



```
int find(int a) { //O(n) ?
    if(tab[a] != a)
        return find(tab[a]);
    return a;
}

void union(int a, int b) { //O(n) ?
    int i = find(a);
    int j = find(b);
    if(size[i] > size[j])
        return union(j, i);
    tab[i] = j;
    size[j] += size[i];
}
```

