Disjoint Sets

Learning Objectives

1. Consider how to implement the Disjoint Set ADT

Disjoint Sets ADT

Maintain a collection S = {s_0, s_1, … s_k}

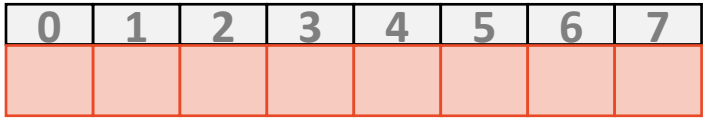
Each set has a representative member.

API: **void makeSets(int number);**

void union(int k1, int k2);

int find(int k);

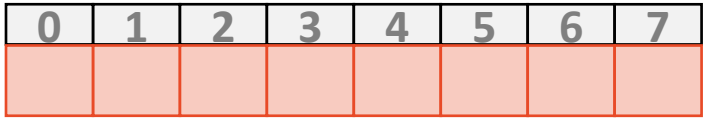
Implementation #1



Find(k):

Union(k1, k2):

Implementation #2



Find(k):

Union(k1, k2):

Implementation #2

0	1	2	3	4	5	6	7

```
1  int DisjointSets::find(int i) {
2    if ( data[i] < 0 ) { return i; }
3    else { return find( data[i] ); }
4  }
```

We will continue to use an array where the index is the key

The value of the array is:

- **-1,** if we have found the representative element
- **The index of the parent**, if we haven't found the rep. element

We will call theses **UpTrees**: