Disjoint Sets
CS 320, Fall 2017

Dr. Geri Georg, Instructor
georg@colostate.edu
Data Structure

Collection of disjoint, dynamic, sets
• Each set has a representative
• Representative is a member of the set
• Can change if set is modified
• Complexity:
  • n, the number of Make-Set operations
  • m, the total number of Make-Set, Union, and Find-Set operations
Operations

\textbf{MAKE-SET}(x) \rightarrow S_x = \{x\}

- x is representative and not in another set

\textbf{UNION}(x, y) \rightarrow S_z = S_x \cup S_y

- representative is any element in either set, original sets “destroyed”

\textbf{FIND-SET}(x) \rightarrow S_x

- x \in S_x

Complexity:

- n objects, & num Make-Set operations
- m, total num Make-Set, Union, and Find-Set operations
Connected Components

**CONNECTED-COMPONENTS** (G)

for each vertex $v \in G.V$

```
MAKE-SET(v)
```

for each edge $(u,v) \in G.E$

```
if FIND-SET(u) \neq FIND-SET(v)
    UNION(u,v)
```

**SAME-COMPONENT** (u,v)

```
if FIND-SET(u) == FIND-SET(v)
    return TRUE
else return FALSE
```

---

$G$ is an **undirected** graph
Linked Lists

Complexity of Make-Set? Find-Set? Union?
Amortized Time

We have $\Theta(n^2)$ time, and $m = 2n - 1$ so we can talk about amortized time:

$$\frac{n^2}{2n - 1}, \text{ so each op on average is } \Theta(n)$$
Trees instead of Lists

Union: Append the smaller lower rank tree as a child of the larger.

Apply path-compression to re-direct child nodes to the root
Image Credits

mazefromdisjsets: http://web.eecs.utk.edu/~plank/plank/classes/cs302/Notes/Disjoint/