Data Structures for Disjoint Sets

謝仁偉 副教授
jenwei@mail.ntust.edu.tw
國立台灣科技大學 資訊工程系
2017 Spring

Overview

- Also known as "union find."
- Maintain collection $\mathcal{S} = \{S_1, \ldots, S_k\}$ of disjoint dynamic (changing over time) sets.
- Each set is identified by a representative, which is some member of the set.
- Doesn't matter which member is the representative, as long as if we ask for the representative twice without modifying the set, we get the same answer both times.

Outline

- ***Disjoint-Set Operations***
- Linked-List Representation of Disjoint Sets
- Disjoint-Set Forests

Operations

- MAKE-SET(x): make a new set $S_i = \{x\}$, and add S_i to $\mathcal{S}$.
- UNION(x, y): if $x \in S_x$, $y \in S_y$, then $\mathcal{S} = \mathcal{S} - S_x - S_y \cup \{S_x \cup S_y\}$.
  - Representative of new set is any member of $S_x \cup S_y$, often the representative of one of S_x and S_y.
  - Destroys S_x and S_y (since sets must be disjoint).
- FIND-SET(x): return representative of set containing x.

Analysis

- Analysis in terms of:
  - n = # of elements = # of MAKE-SET operations,
  - m = total # of operations.
- Since MAKE-SET counts toward total # of operations, $m \geq n$.
- Can have at most $n - 1$ UNION operations, since after $n - 1$ UNIONs, only 1 set remains.
- Assume that the first n operations are MAKE-SET (helpful for analysis, usually not really necessary).

Application (1/2)

- Dynamic connected components:
  - For a graph $G = (V, E)$, vertices u, v are in the same connected component if and only if there's a path between them.
  - Connected components partition vertices into equivalence classes.

```
CONNECTED-COMPONENTS(G)
for each vertex v ∈ G.V
    MAKE-SET(v)
for each edge (u, v) ∈ G.E
    if FIND-SET(u) ≠ FIND-SET(v)
        UNION(u, v)
```


Application (2/2)

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


Outline

- Disjoint-Set Operations
- ***Linked-List Representation of Disjoint Sets***
- Disjoint-Set Forests