Union-Find
Components
Union-Find ADT
Algorithms
Disjoint Sets
Make/Union
Find-Set
Speed-Ups
Ackermann

Union-Find

Connected Components

Problem: find components in an undirected graph and answer membership queries

How can we identify components?

Union-Find ADT

Union-Find
Components
Union-Find ADT
Algorithms
Disjoint Sets
Make/Union
Find-Set
Speed-Ups
Ackermann

MAKE-SET(x): makes a new set containing only x

UNION(x, y): combines the set containing x with the set containing y

FIND-SET(x): returns a representative of the set containing x

Connected Components Algorithms

Union-Find
Components
Union-Find ADT
Algorithms
Disjoint Sets
Make/Union
Find-Set
Speed-Ups
Ackermann

BUILD-COMPONENTS(G)

```
1: for all vertex v do
2:     MAKE-SET(v)
3: for all edge (u, v) do
4:     UNION(u, v)
```

IS-SAME-COMPONENT(u, v)

```
1: return FIND-SET(u) == FIND-SET(v)
```

Union-Find
Components
Union-Find ADT
Algorithms
Disjoint Sets
Make/Union
Find-Set
Speed-Ups
Ackermann

Disjoint Sets

A set is a tree rooted at a representative

How do we implement make/union/find efficiently?

MAKE-SET, UNION Algorithms

MAKE-SET(x)

```
1: x.π ← x
2: x.rank ← 0
```

UNION(u, v)

```
1: x ← FIND-SET(u)
2: y ← FIND-SET(v)
3: if x.rank > y.rank then
4:     y.π ← x
5: else
6:     x.π ← y
7:     if x.rank = y.rank then
8:         increment y.rank
```

Union-Find
Components
Union-Find ADT
Algorithms
Disjoint Sets
Make/Union
Find-Set
Speed-Ups
Ackermann

FIND-SET Algorithm

FIND-SET(x)

1: **if** $x \neq x.\pi$ **then**
2: $\quad x.\pi \leftarrow$ FIND-SET$(x.\pi)$
3: return $x.\pi$

Speed-Ups

union by rank: Track tree height, put shorter trees under taller ones

path compression: after FIND-SET, ensure touched nodes point directly to root

For m operations on n sets, worst-case runtime is $O(m\alpha(n))$

Inverse Ackermann Function

$\alpha(n)$ is the inverse Ackermann function:

- $\alpha(3) = 1$
- $\alpha(7) = 2$
- $\alpha(2047) = 3$
- $\alpha(3231700607131100730071487668866995196044410266971548403213034542752465513886789089319720141152291346368871796092189801949411955915049092109508815238644828312063087367300996091750197750389652106796057638384067568276792218642619756161838094338476170470581645852036305042887575891541065808607552399123930385521914333389668342420684974786564569494856176035326322058077805659331026192708460314150258592864177116725943603718461857357598351152301645904403697613233287231227125684710820209725157101726931323469678542580656697935045997268352998638215525166389437335543602135433229604645318478604952148193555853611059596230656) = 4$

So... we usually treat $\alpha(n)$ as a constant, but we acknowledge it.