March 10

Union-Find.

Given n elements $x_1, x_2, \cdots, x_n$

optimize:

(1) make set $(x) \longrightarrow \{x\}$

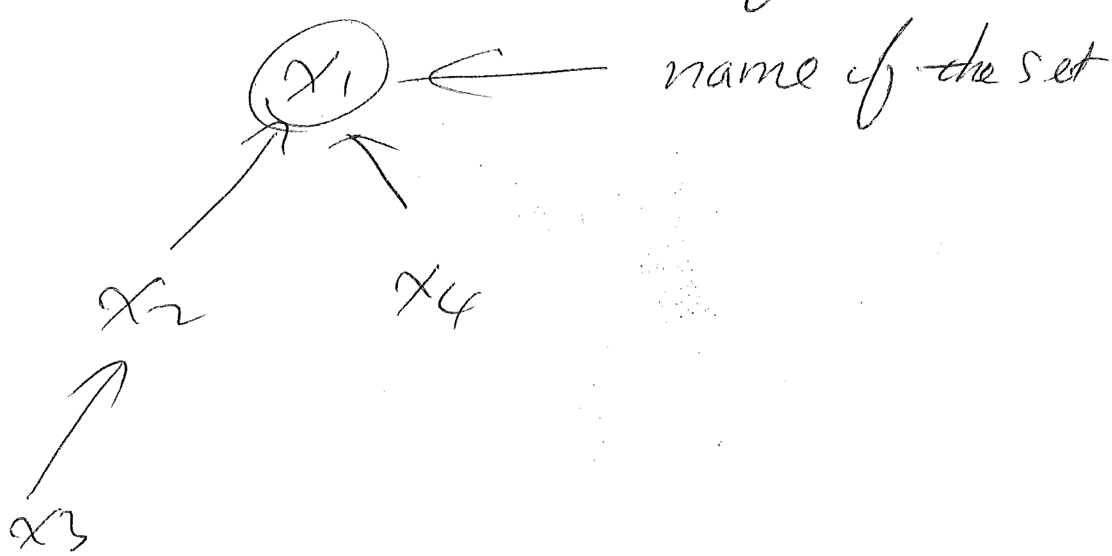
(2) find set $(x) \longrightarrow$ return the name of the set containing x

(3) union $(x, y) \longrightarrow$ merge the sets containing x and the set ~~and~~ containing y

Strategy.

(1) use trees to represent a set

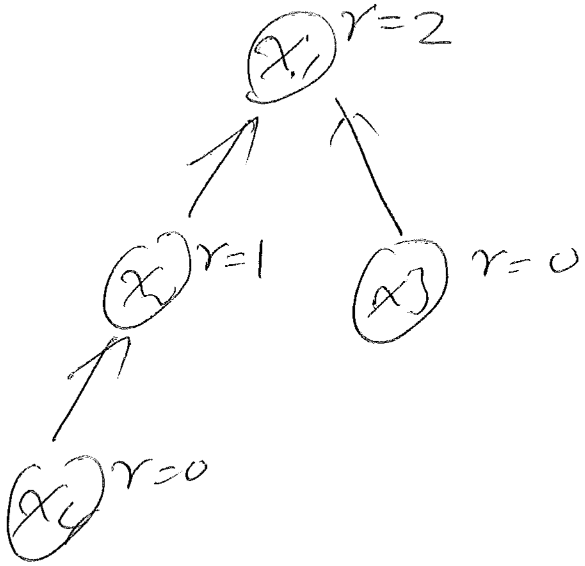
the root is the name of the set



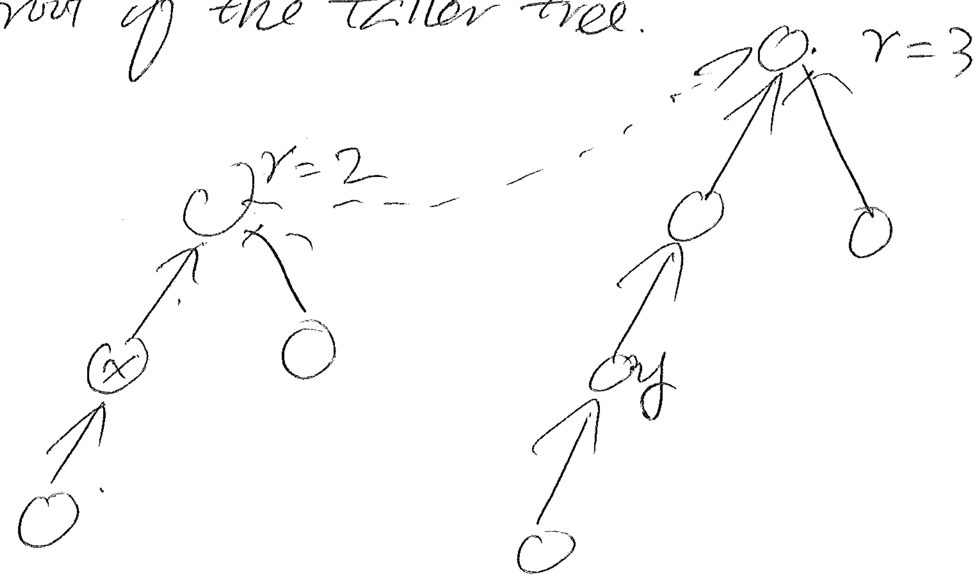
(2) Union by rank

(control the height of the tree)

For each node, we will associate an attribute called "rank" which is the height of its subtree



union-by-rank when union (x, y) we will connect the root of the shorter tree to the root of the taller tree.



With union-by-rank, find and union are $O(\log n)$ operations

~~Proof (By induction on the rank k)~~

Observation: the number of nodes ever reaching rank k is bounded by $\frac{n}{2^k}$

The number of node that can reach rank $\log n$ is $\leq \frac{n}{2^{\log n}} = 1$

$\log n$ is the maximum height of any tree in the data structure

Proof by induction on k

Basis $k=0$

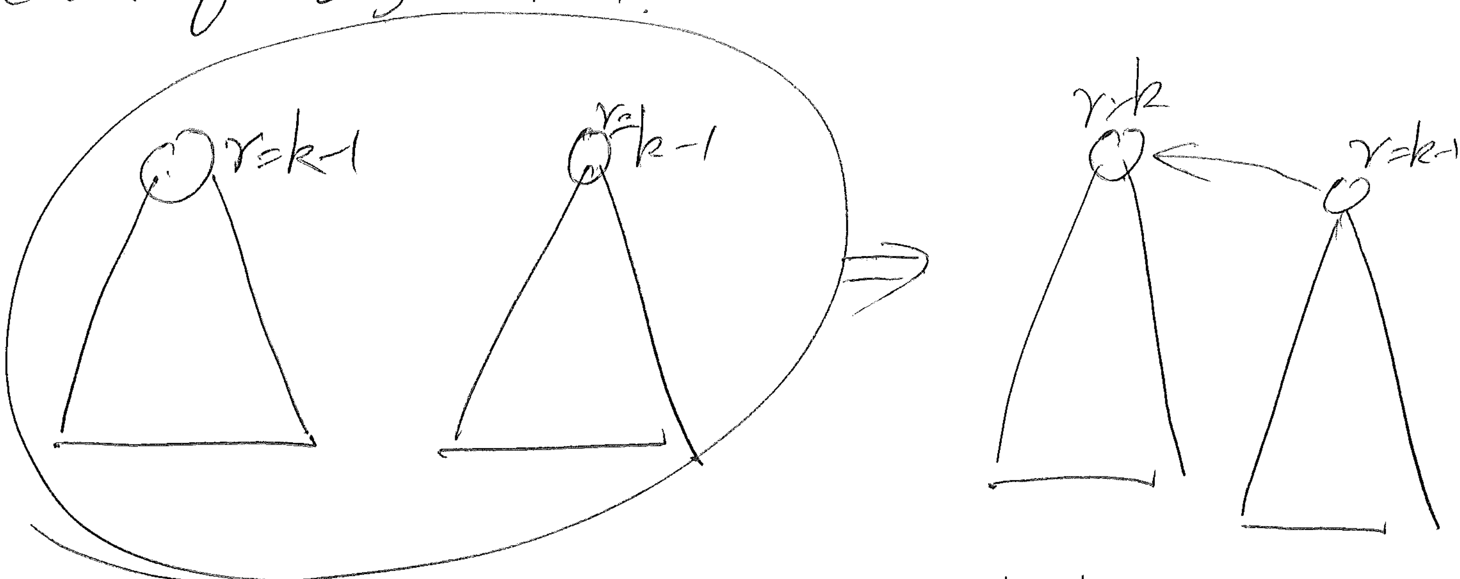
Observe that the maximum number of nodes that can have rank 0 is $\leq n$

$$\frac{n}{2^0} = n$$

I.H. Assume the claim correct for rank $k-1$
the number of nodes with rank $k-1$ is $\leq \frac{n}{2^{k-1}}$

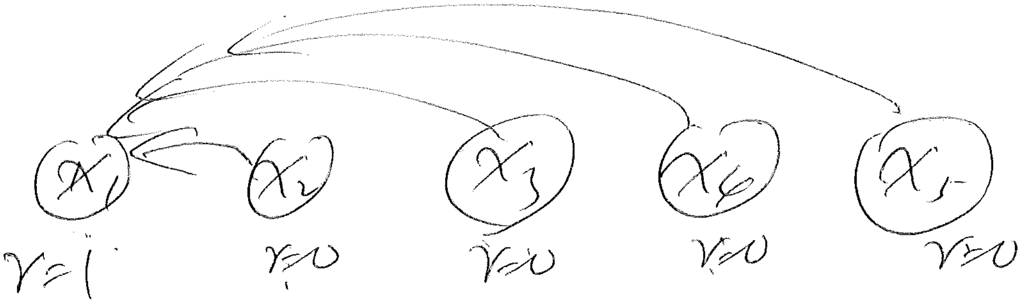
I.S. Need to show the case of k.

Observe that the only way to create a
rank k node is when union two trees
each of height $k-1$.



Thus the number of nodes with rank k is
at most $\frac{1}{2}$ of the number of nodes with rank $k-1$

i.e., $\frac{1}{2} \cdot \frac{n}{2^{k-1}} = \frac{n}{2^k}$



union (x_1, x_2)

union (x_2, x_3)

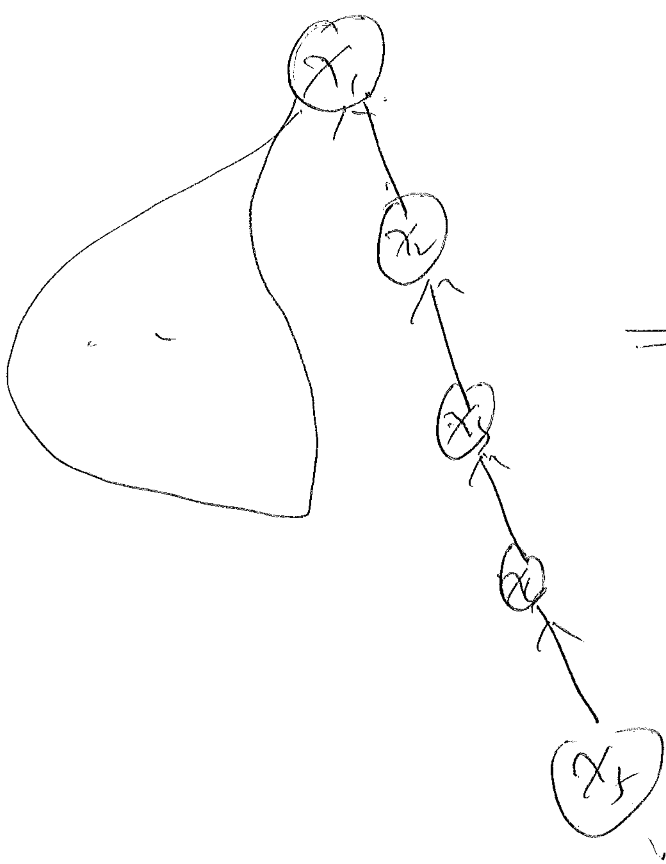
union (x_2, x_4)

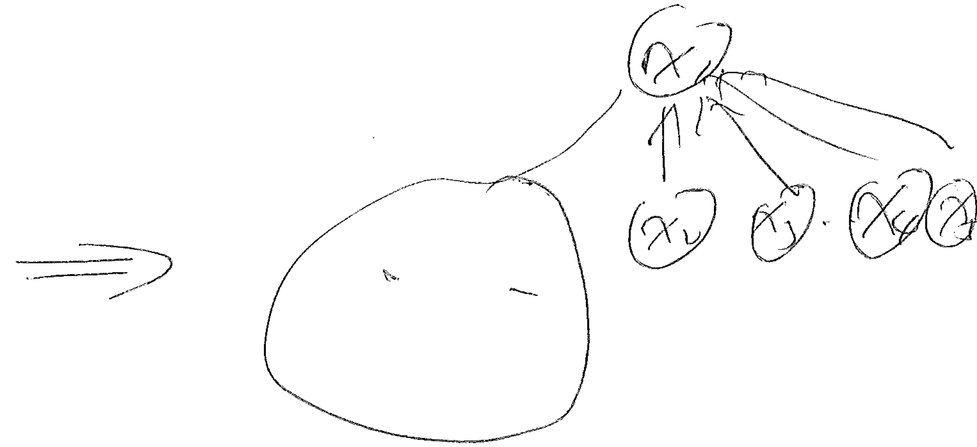
union (x_2, x_5)

(3) Path Compression

Find operations is bounded by tree height

find (x_5)





With path compression,

for m find operations, the running time

$(m+n) \log^* n$

Tower function: $2^{2^{2^{\cdot^{\cdot^{2}}}}}\Big\}n$

Tower(n)

Example:

$\text{Tower}(3) = 2^{2^2} = 2^4 = 16$

$\text{Tower}(4) = 2^{2^{2^2}} = 2^{16} = 64k$

$\text{Tower}(5) = 2^{2^{2^{2^2}}} = 2^{64k}$

$\log^* n$ is the number of log operations to

bring n down to ≤ 1

$$\log 2^{2^{2^{2^{2}}}} = 2^{2^{2^{2}}} \cancelto{1}{\log 2}$$

$$\log 2^{2^{2^{2}}} = 2^{2^{2}} \cancelto{1}{\log 2}$$

$$\log 2^{2^{2}} = 2^{2}$$

$$\log 2^{2} = 2$$

$$\log 2 = 1$$

$$\log^{*} 2^{2^{2^{2^{2}}}} = 5$$

Kruskal's Algorithm

input $G(V,E)$

output MST : associate an attribute to each edge
if it's marked, then edge belongs to the MST

Unmark all edges

$\boxed{\text{For all } v \in V, \text{ make set}(v) \quad \text{unmark all edges}}$

$|E| \log |V|$

put all edges in a min-heap H based on the edge costs

While H is not empty

~~e = extra~~ $e(u,v) = \text{extract\_root}(H)$

if $\text{find}(u) \neq \text{find}(v)$

union(u,v)

mark(e)

Running Time

$$|E| \log |E| \leq |E| \log (|V|^2) = O(|E| \log |V|)$$

Randomized Algorithm.

Quick Sort.

input: ~~an array~~ a set S of n numbers, $S = \{s_1, s_2, \cdots, s_n\}$

output S sorted.

Pick a pivot x. → Find the medium x.

use x to partition S into $S_1 \leq x < S_2$

Recursively perform quick sort on S_1 and S_2

$$T(n) = n + T(|S_1|) + T(|S_2|)$$

$$= n + T(k) + T(n-k) \quad \text{for } 1 \leq k \leq n$$

if $k = 1$ $\quad T(n) = n + T(n-1) \Rightarrow T(n) = O(n^2)$

$k = \frac{n}{2}$ $\quad T(n) = 2T(\frac{n}{2}) + n \Rightarrow T(n) = O(n \lg n)$

Randomized Quick Sort

<u>randomly</u> pick an element x as the pivot

use x to partition $S_1' \le x < S_2'$

recursively sort S_1' and S_2'

the <u>expected</u> running time : $O(n \log n)$

Propabilitistic experiment.

experiments whose outcome are not deterministic and subject to some intrinsic ~~probability~~ diseribution

Probability Space

S Sample Space: the set of all the outcomes

$\mathbb{E}$: Event Set. the collection of events of interest (2^S)

P: probability function. $P: \mathbb{E} \longrightarrow [0,1]$

$$0 \leq P(E) \leq 1$$

Consider tossing a 6-sided die

e.g. $S' = \{1, 2, 3, 4, 5, 6\}$

$\mathbb{E}$: an event is a subset of the sample space

an event happens if the outcome of a particular experiment belongs to the event

The event set $= 2^{S'}$

$P: \mathbb{E} \to [0, 1]$

elementary event: each individual outcome as an event $\{1\}$ $\{2\}$ $\{3\}$ $\{4\}$ $\{5\}$ $\{6\}$

elementary probability: probabilities of elementary events

$P(\{1\}) = P(\{2\}) = P(\{3\}) = P(\{4\}) = P(\{5\}) = P(\{6\}) = \frac{1}{6}$

equally likely outcomes $\Longleftrightarrow$ fair die

Axioms of Probability.

(1) $0 \leq P(Z) \leq 1$

certain event

(2) $P(\phi) = 0$, $P(S) = 1$

impossible event

(3) For $Z, F \in \mathbb{E}$, $Z \cap F = \phi$

$P(Z \cup F) = P(Z) + P(F)$

(4) $\mathbb{E}$ is closed under union, intersection, complement

$$P(\{2,4,6\}) = P(\{2\}) + P(\{4,6\})$$

$$= \underline{P(\{2\})} + \underline{P(\{4\})} + \underline{P(\{6\})}$$

$$= \frac{1}{6} + \frac{1}{6} + \frac{1}{6} = \frac{1}{2}$$

Define (Joint Probability)

Let E, F be two events, their joint probability ~~of~~ is $P(E \cap F)$

Define (Independence)

Two events E, F are independent if

$$P(E \cap F) = P(E) \cdot P(F)$$

Example: Considering tossing a fair 6-sided die

$E = \{2,4,6\}$ $\quad$ $F = \{3,4,5\}$

are E and F independent?

$$P(E \cap F) = \frac{1}{6} \qquad P(E) = \frac{1}{2} \qquad P(F) = \frac{1}{2}$$

$$P(E \cap F) \neq P(E) \cdot P(F)$$

$E = \{2,4,6\}$ $\quad$ $F = \{3,4\}$

$$P(E \cap F) = \frac{1}{6} \qquad P(E) = \frac{1}{2} \qquad P(F) = \frac{1}{3}$$

Def (Conditional Probability)

$$P(E|F) = \frac{P(E \cap F)}{P(F)}$$

$E = \{2,4,6\}$ $\quad$ $F = \{3,4,5\}$

$$P(F|E) = \frac{\{4\}}{\{2,4,6\}} = \frac{1}{3}$$

$$= \frac{P(E \cap F)}{P(F)} = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$$