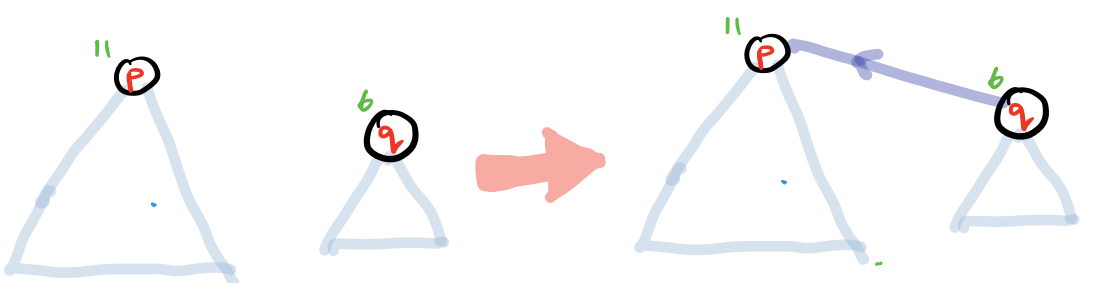
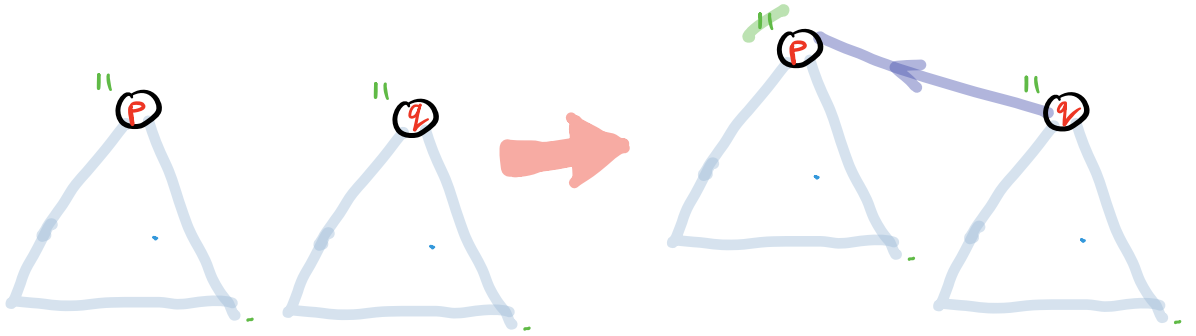
Disjoint-Sets ADT Union-Find

Union-by-rank

- Find the roots of your arguments first
- The rank of an singleton set is 0.
- If sets of p of rank $r+\Delta$ where $\Delta \geq 0$ and q of rank r are unioned, q becomes the child of p



if $\Delta > 0$, no rank needs updating



- if $\Delta = 0$ (i.e. same rank) the first argument p's tree becomes the root

Theorem: If a root (in a union-find forest that uses union-by-rank) has rank r then it has $\geq 2^r$ nodes in its tree.

Proof: By induction on r.

Basis: Suppose a root p has rank $r=0$
It has ≥ 1 element in its tree,
so has $\geq 2^0 = 2^r$ nodes in its tree

Inductive case: Suppose root p has rank $r>0$,
and it had its rank increased to r when
it was unioned with tree with root q.

Ind. Hyp: trees with roots of rank $<r$ have the property.

p and q both had rank $r-1$ at the time they were unioned.

$\Rightarrow$ p's tree had $\geq 2^{r-1}$ nodes, by Ind Hyp, and
q's tree had $\geq 2^{r-1}$ nodes, by Ind Hyp

$\Rightarrow$ after the union, p's tree has $\geq 2 \cdot 2^{r-1}$ nodes,
i.e. has $\geq 2^r$ nodes. ▨

Theorem: If a root (in a union-find forest that uses union-by-rank) has rank r then it has height $\leq r$

Proof: By induction on r.

Basis: Suppose a root p has rank $r=0$
Then it has never been unioned,
and has height $0 = r$.

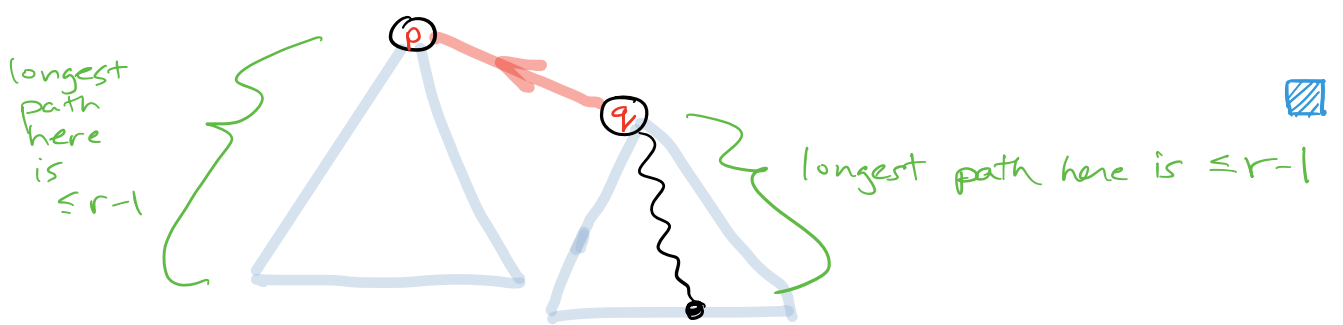
Inductive case: Suppose root p has rank $r > 0$,
and it had its rank increased to r when
it was unioned with tree with root q.

Ind. Hyp: trees with roots of rank $< r$ have the property.

p and q both had rank $r-1$ at the time they were unioned.

$\Rightarrow$ p's tree had height $\leq r-1$ by Ind Hyp, and
q's tree had height $\leq r-1$ by Ind Hyp

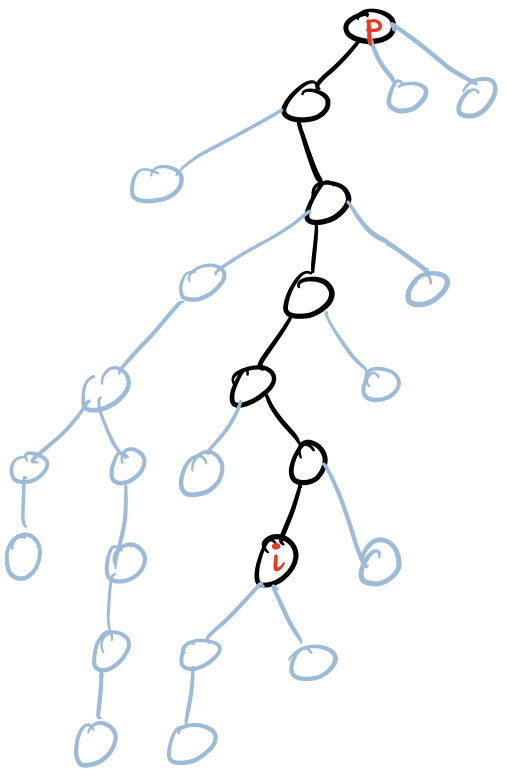
$\Rightarrow$ after the union, p's tree has height
$\leq r$



∎

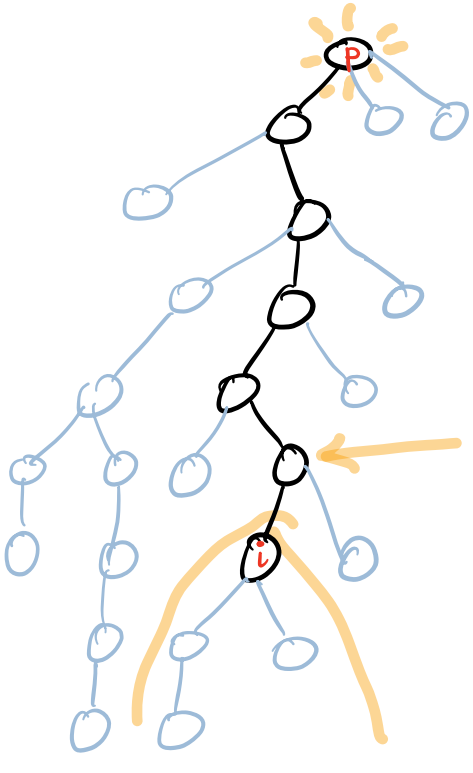
To keep our trees short and bushy...

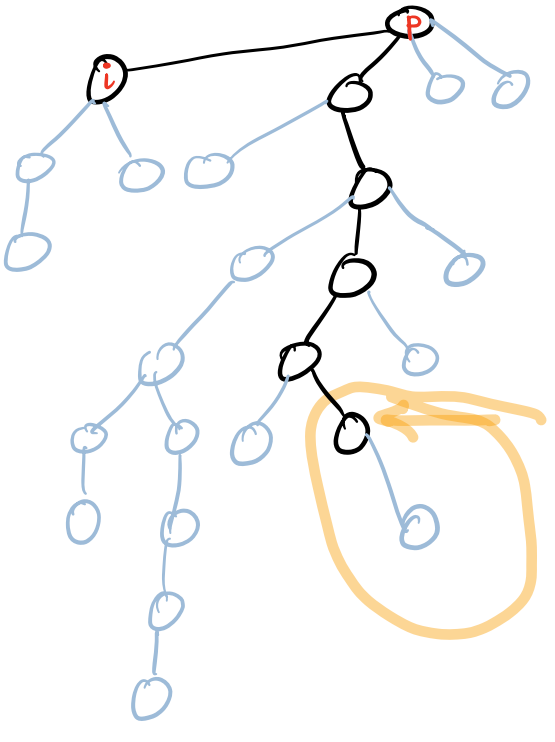
When we do a find(i), we know we will be traversing the entire path up from i to its root. Let's say that is K hops.

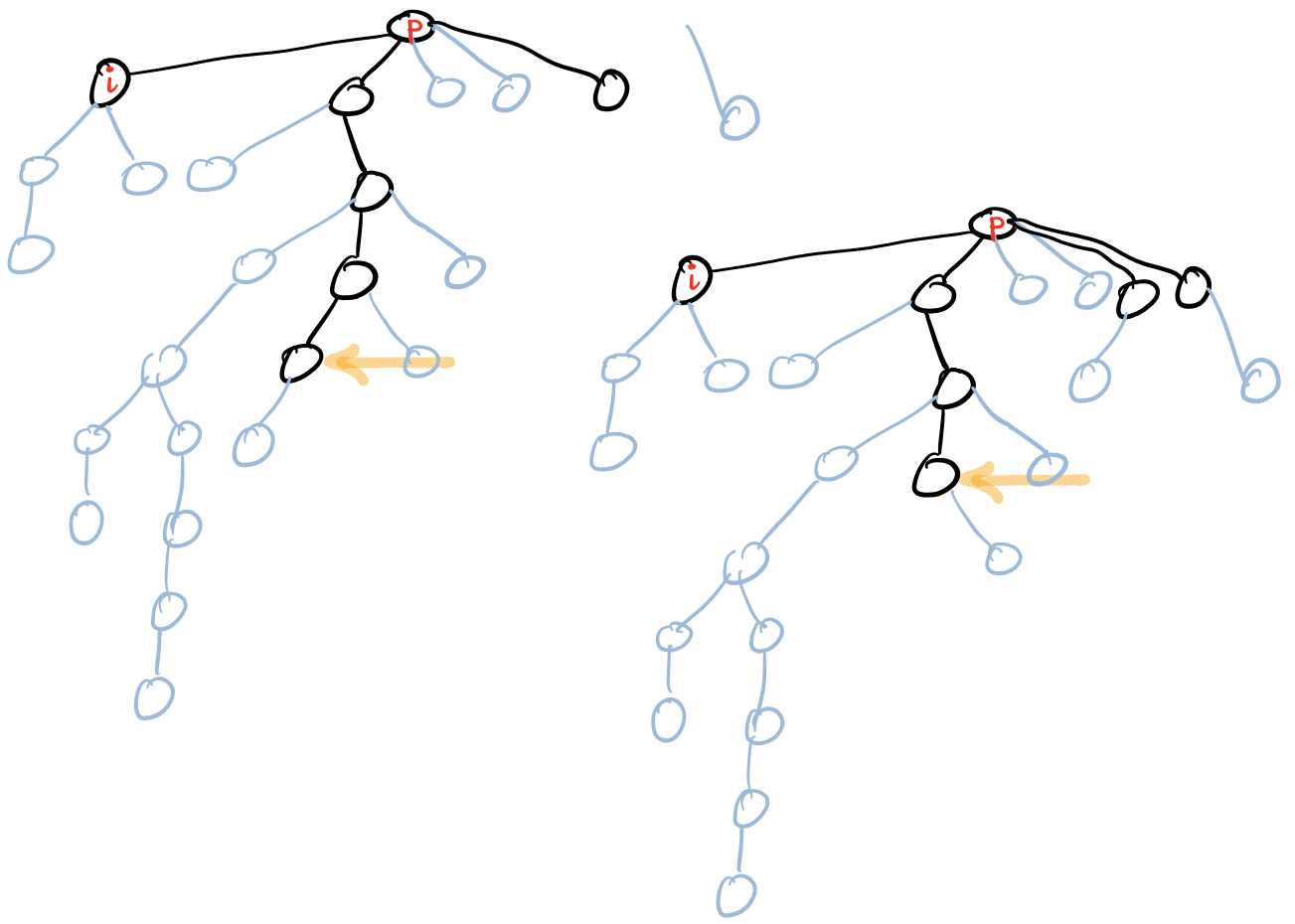


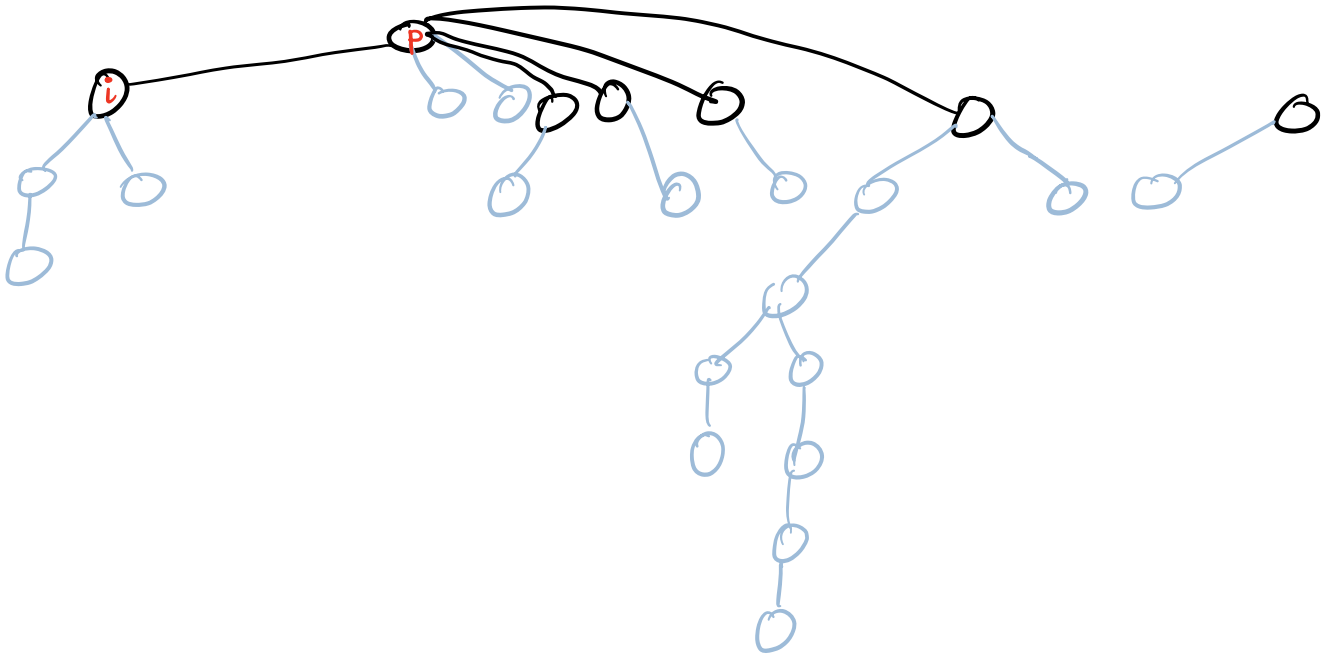
How much compute time does that take?

O(K)






P
i
P
i


P
i