day 7 2/2/98 "1, 2, 3, 4, 5, 6, 7, 8, 9. There should be 9 of these, right?"

1 Treaps

- keys $\{1, ..., n\}$
- priority $p(k) \in [0, 1]$. if $k \neq l$ then $p(k) \neq p(l)$
- T a tree with a key at each node.
- T in heap order = $\forall x, y \in T : x = parent(y)\ p(x) < p(y)$

Lemma: A heap order binary search tree always exists and is unique. proof: keys must satisfy in order constraint and heap ordering constraint. $\exists$a smallest (heap order) element. This must be the root. Every smaller (by key order) element must be to the left of the root and larger (by key order) elements must be to the right. Now induct ont he left and right subtrees.

2 Random Treap

A random treap has priorities of each of its elements assigned randomly.

2.1 insert(k)

1. insert k as a leaf
2. pick $p(k)$ randomly
3. rotate k up till you are in heap order

2.2 delete(k)

1. rotate k down to a leaf
2. remove k

2.3 Search(k)

Search(k) is implemented in the same way as for other binary search trees.

2.4 Search Timing analyses

Let:

- C = total number of comparisons
- C^+ = total number of m>k comparisons

- C^- = total number of m<k comparisons

Break the expected cost of a search is:

$E_C(m) = E_{C^+}(m) + E_{C^-}(m)$

Consider searching for $m + .5$. Then C^+ will contain elements only from $\{1, ..., m\}$. and C^- will contain elements only from $\{m+1, ..., n\}$. The probability that k in C^+ is chosen will be $\frac{1}{m-k+1}$. $\sum_{i=1}^{m} \frac{1}{m-i+1} = \Theta(ln(m))$. Calculate the contribution from C^- in the same way, then add them together to get $E_C(m) = \Theta(ln(n))$.

2.5 Deletion timing analyses

In order to delete, elements must be rotated to the leaves before removal. The expected number of rotations is less than 2. This can be seen by noting that rotations will follow the left most branch of the right child or the right most branch of the left child, then calculating the odds of a long left most or right most chain. Essentially, the tree is probabilistically "fat" in that almost all nodes are near to a leaf.