Announcements

- Test Friday (review Thurs.)
- HW due tonight
  - Next HW - up
  - checkpoint in 1 week
Note on HW: #5

1, 2

2, 1
Treaps: a new binary tree data structure

- Nodes will contain both values and priorities

- A treap is a BST over the values and a heap over the priorities.
Example:

```
    M
   / 
  H/2 T/3
 /   / 
G/7 E/4 R/5
 /   / 
A/9 F/8
```

data: BST
priority: heap
Insert

insert: (5, 0)
start BST: ✓
heap?
In heap we "bubbled up". Will that work here?

No: bubbling up may break BST field.
What to do?
Rotations

$x \geq y$ are in correct BST order, with $x \leq y$, but priorities are wrong.

Fix:

- Pivot (from Binary Tree class)
So: insert (S, 0)
Downside: What can length be? \( O(n) \)

Can we force them to be balanced? No
Draw heap with \((A, 4), (C, 2), (X, 11), (M, 3), (Q, 1), (Z, 5)\)
Randomized treaps:

Alternative to AVL trees.
Each element will get a random priority.
Expected height of the treap will be \(O(\log n)\).
Code: How do we implement?

- Inherit from BinaryTree.h (since not complete tree, like heap)
- Code is actually easier than AVL

while (priority out of order)

Next time: implement