Lec 11 Balanced Binary Tree I

Useful for
- ordered sets
- ordered tables
- sequences
- ⋮

Seen
- Remove, insert, find,
- AVL
- Redblack
- BSTs (binary search tree)

More bin tree ops

- insertAt *
- deleteAt *
- nth
- intersect
- union
- difference
- append *
- ranges
- split
- map
- reduce
- filter

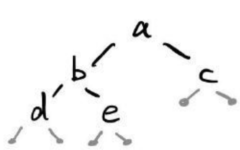
* will be better than ArraySeq

<u>Tree options</u>

- AVL
- Red Black
- Treaps
- Splay
- BTree
- Skapegoat
- Weight balanced
- 2-3 tree
- Skip-list

So many of them. Want to abstract all the options.
Assume there's joinMid for each tree type, implement general operations.

<u>Binary tree</u>



← "internal binary tree" as we don't store data on the leaves

<u>Def</u> Balanced := height $\in O(\log n)$
usually height $\leq 2\lg n$

Note this is always true:
height $\geq \lceil \lg(n+1) \rceil$
height $= \lceil \lg(n+1) \rceil$ when perfectly balanced

<u>Store at nodes</u>

- Value
- Key
- Balancing info
- Size of subtree
- Associative info (augmentation)

Binary Search Trees

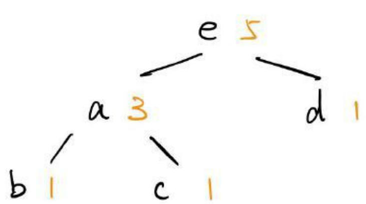
Def $\forall_{node}, \begin{cases} \forall k \in \text{Left}, & k < \text{root} \\ \forall k \in \text{Right}, & \text{root} < k \end{cases}$

Sequence Tree

Binary tree + size of subtree

Inorder traversal of tree is the sequence

$\langle b, a, c, e, d \rangle$ sizes



Exposing : get rid of extra info and return barebone tree