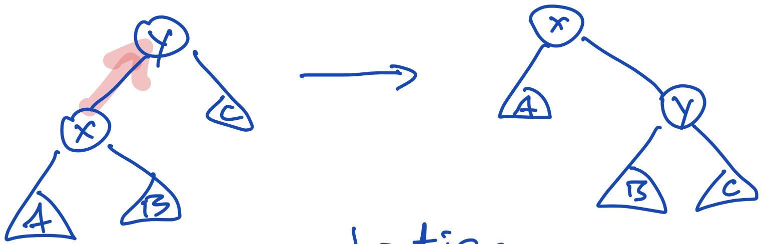
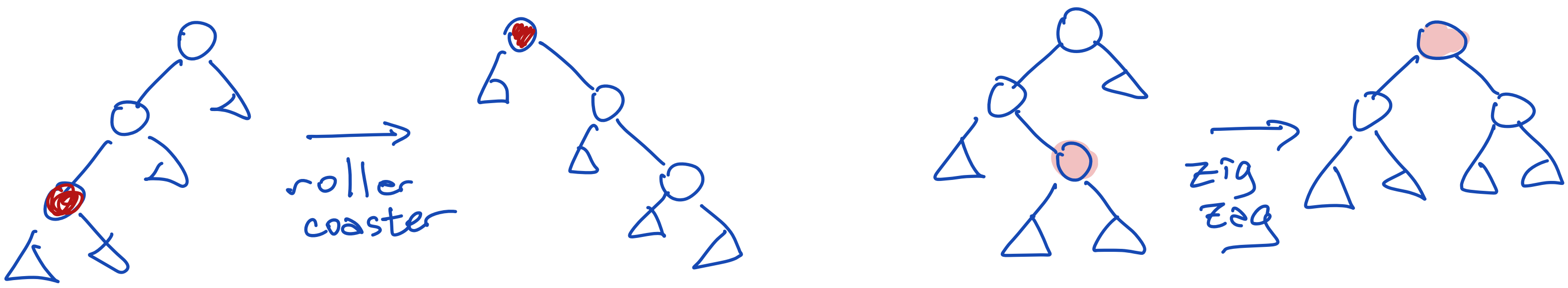
BST — ordered dictionary

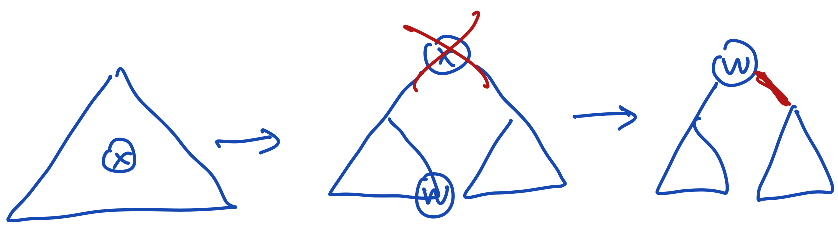


rotation

double rotations



Splay(x)
  while parent(x) ≠ root and x ≠ root
    double rotate(x)
  if x ≠ root
    rotate(x)



Time for
find, suc, pred,
ins, del

$= O(\text{splay}) = O(\text{depth}(x))$

Amortized time: starting with empty/balanced

Theorem: Splay executes any sequence of N splays in $O(N \log n)$ time

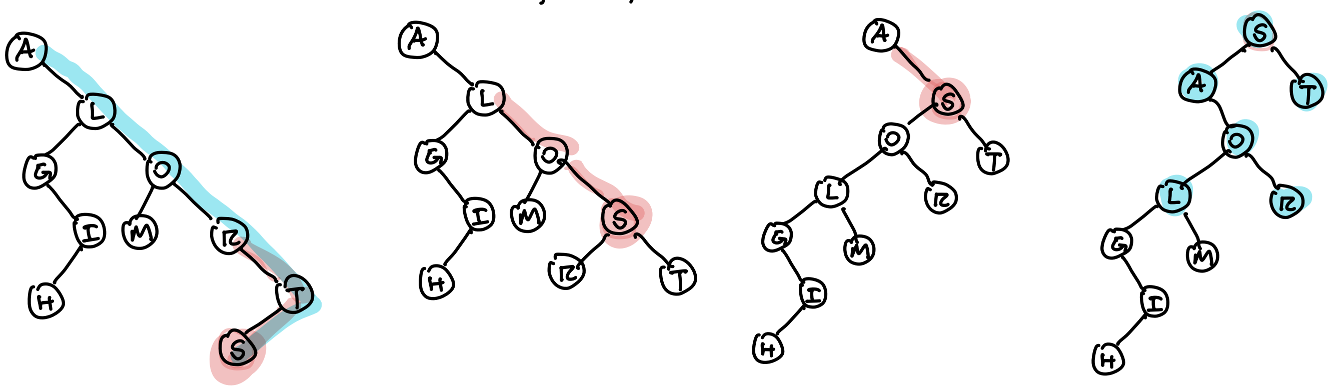
Proof: potential method

$\text{size}(v) = \#\text{descendants of } v$

$\text{rank}(v) = \lfloor \lg \text{size}(v) \rfloor$

$\Phi(T) = \sum_v \text{rank}(v)$

Splay Trees



Amortized time(op) := Time(op) + $\Phi_{new} - \Phi_{old}$

$$\sum_{op} AT(op) = \sum_{op} T(op) + \Phi_{final} - \cancel{\Phi_{init}}$$

$$\sum T(op) \leq \sum AT(op)$$

Access Lemma: [Sleator Tarjan 85]

$$\boxed{\begin{array}{ll} \text{Am. time to rotate } v & \leq 1 + 3\,\text{rank}'(v) - 3\,\text{rank}(v) \\ \text{Am time to dbl-rot } v & \leq 3\,\text{rank}'(v) - 3\,\text{rank}(v) \end{array}}$$

$\Rightarrow$ Am. time to splay $v \leq 1 + 3\,\text{rank}_{final}(v) - 3\,\text{rank}_{init}(v)$

$\leq 1 + 3\,\text{rank}_{final}(v)$

$\leq 1 + 3 \lg n$

Suppose we search for x $t(x)$ times $\quad T = \sum_x t(x)$

Access Lemma still works if $\text{size}(v) = \sum_{x \Downarrow v} t(x)$

$\text{rank}(v) = \lg \text{size}$

$\Phi = \sum \text{rank}$

$\Rightarrow$ Am time to splay $x = 1 + 3 \cdot \lg T - 3\,t(x)$

$= 1 + 3 \lg \frac{T}{t(x)}$

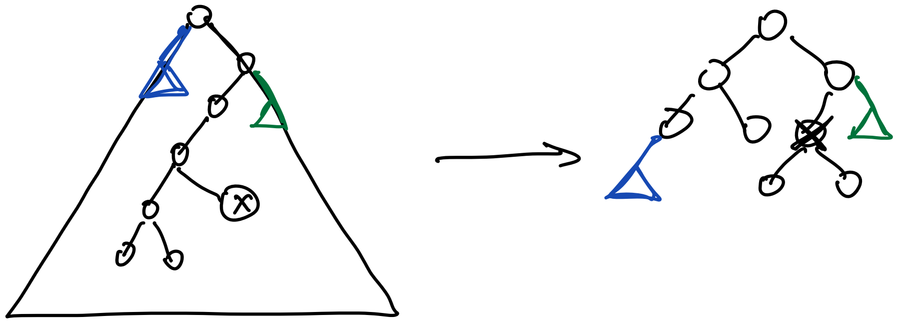
Static optimality

Conjecture: splay trees are dynamically opt

$O(1)$-competitive ratio

vs. best offline $\boxed{\text{dynamic BST}}$

Dynamic BST?



R = some upward-closed subset of nodes includes x

arbitrarily reconfigure R in $O(|R|)$ time

splay trees

R = search path

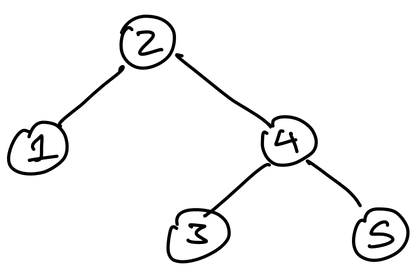
splay

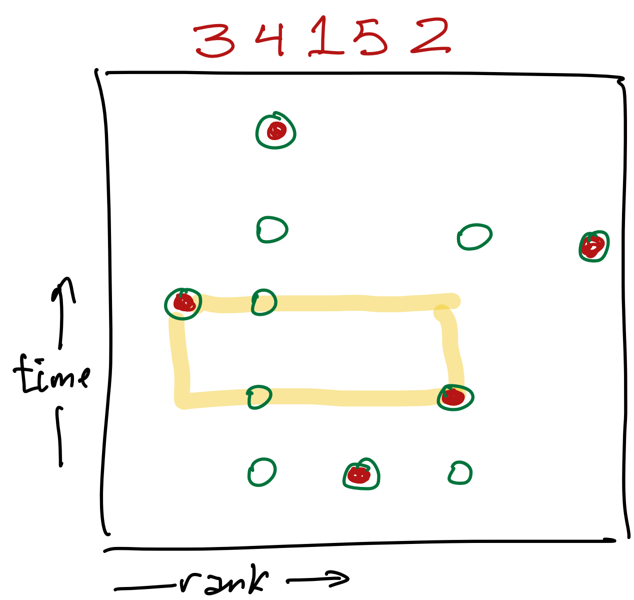
Weaker Conjecture: There is an $O(1)$-competitive dynamic BST

Geometric View of BSTs: [Demaine Harmon Iacono Kane Pătraşcu]

Access sequence $x_1, x_2, \ldots x_N \Rightarrow (x_i, i)$

Execution sequence = $\{(y, i) \mid$ node y is touched during ith access$\}$



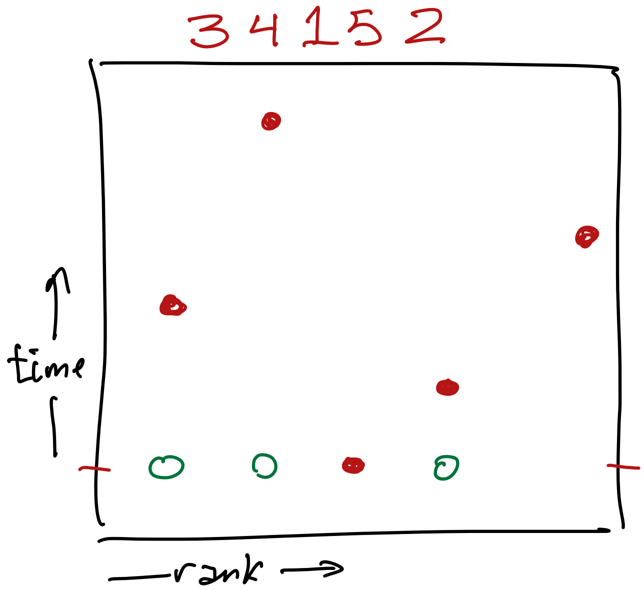


Thm: Execution pts are satisfied meaning every rect defined by two pts has another pt on it's boundary

Thm: And vice versa.

BADNEWS: Finding min satisfying superset is NP-hard (if we allow multi-accesses)

Natural Greedy heuristic:



Greedy Future:

For $i \leftarrow 1$ to N
add min #pts on row i
to satisfy rectangles
with bottom row i

$O(N \log n)$ time

Conj: GREEDY FUTURE
is $O(1)$-competitive

BST language — optimally reconfigure only search path

Greedy Past is an online algorithm → GREEDY BST

$\text{Greedy} \geq \text{OPT} \geq \max\{\text{Greedy}_{\boxtimes}, \text{greedy}_{\boxtimes}\}$

Greedy

Greedy

Conj: $\text{Greedy} \leq \text{OPT} + O(1)$