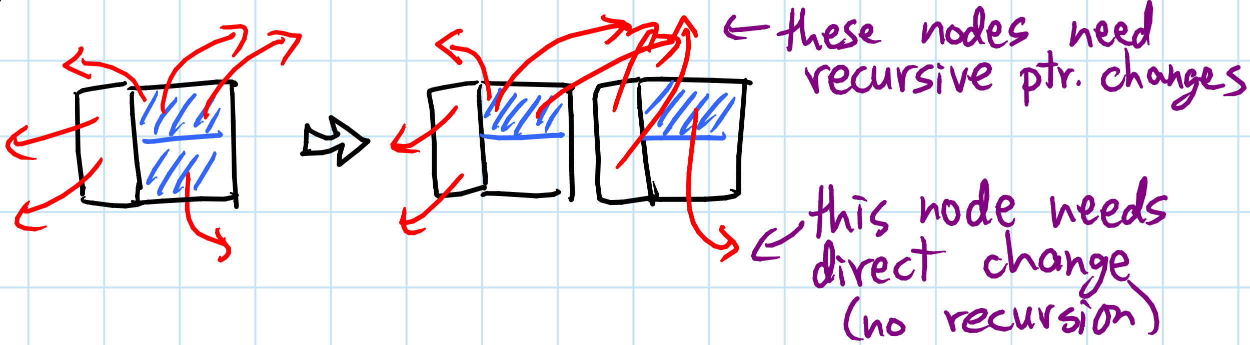
6.851	Class 3	Feb. 19, 2014

TODAY: geometric DS
- full persistence amortization fix (quick)
- fractional cascading example
- 3D orthog. range search top-down
- kinetic survey

Full persistence amortization: fix
- key: pointers in d+p+1 mods (of both nodes) have reverse pointers that can be updated directly to just one of the nodes

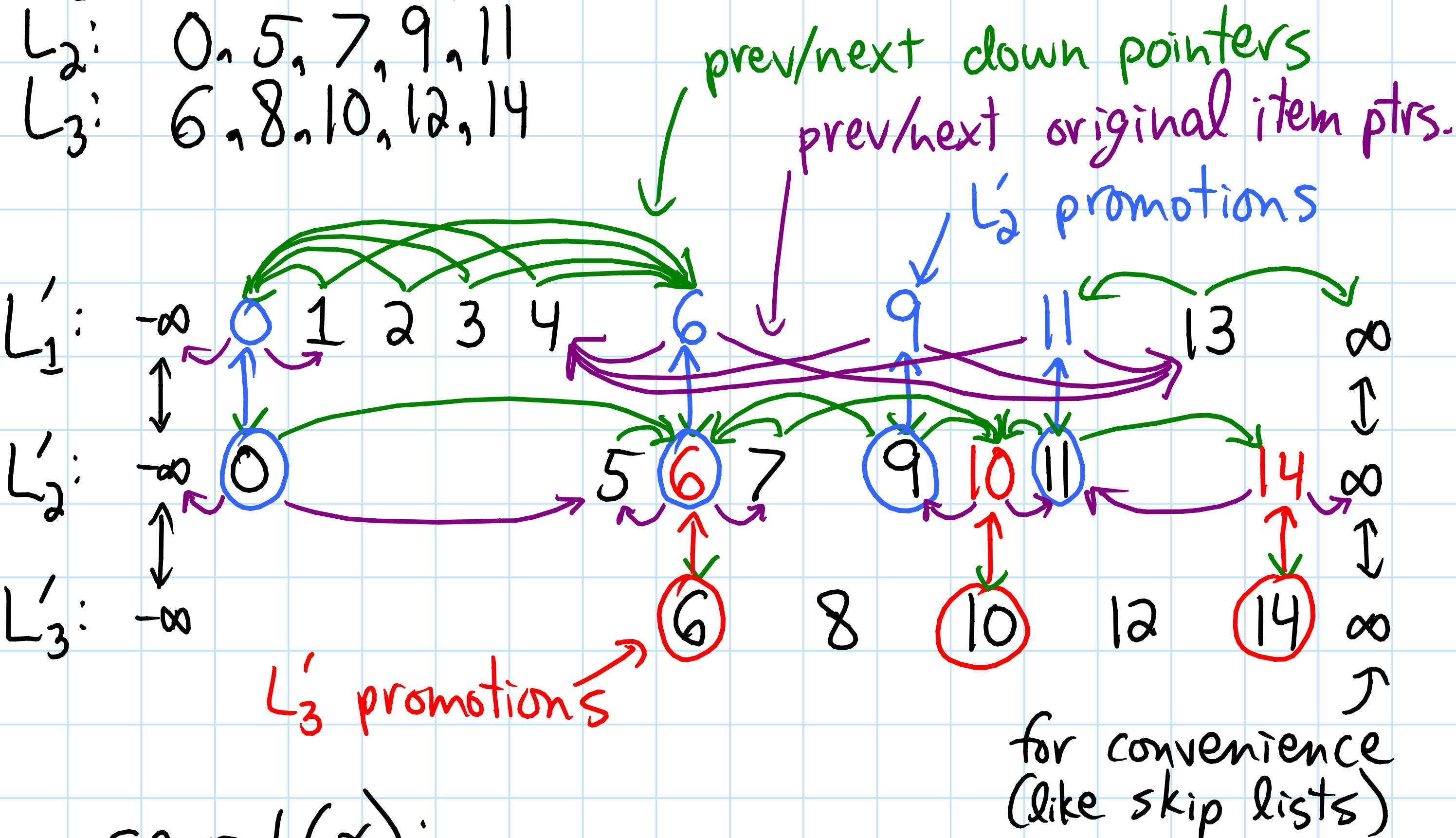


STUDENT: "Fractional cascading was the coolest thing I've learned about in a while — the bound achieved doesn't seem like it should be possible."

Fractional cascading example: (static)

predecessor/successor(x) in k sorted lists

L_1: 1, 2, 3, 4, 13
L_2: 0, 5, 7, 9, 11
L_3: 6, 8, 10, 12, 14



search(x):
- binary search in L_1' for pred/succ. of x
- go to prev & next down pointer
- walk left/right 0 or 1 steps to predecessor/successor of x in L_i'
- pred./succ. in L_i via original item ptrs.
- repeat (i += 1) from pred/succ. in L_i' (not L_i)

e.g. x = 5, 9, 12

3D range queries in $O(\lg n + k)$: top-down review

- static
- reporting only: k = # desired pts. in box

- "range tree" on z:
  - query: $[a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$
  - leaves = points in z order
  - walk down to $\mathrm{lca}(a_3, b_3) = v$ $+O(\lg n)$
  - reduce to 2 queries at node v:

    $[a_1, b_1] \times [a_2, b_2] \times (a_3, \infty)$ in left(v)

    $\cup$ $[a_1, b_1] \times [a_2, b_2] \times (-\infty, b_3)$ in right(v)

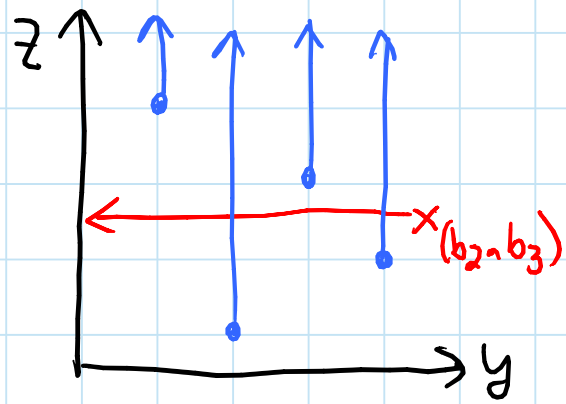
  $\Rightarrow$ each node stores the following on subtree:
- "range tree" on y:
  - query: $[a_1, b_1] \times [a_2, b_2] \times (-\infty, b_3)$ & symmetric
  - reduce to 2 queries at node v': $+O(\lg n)$

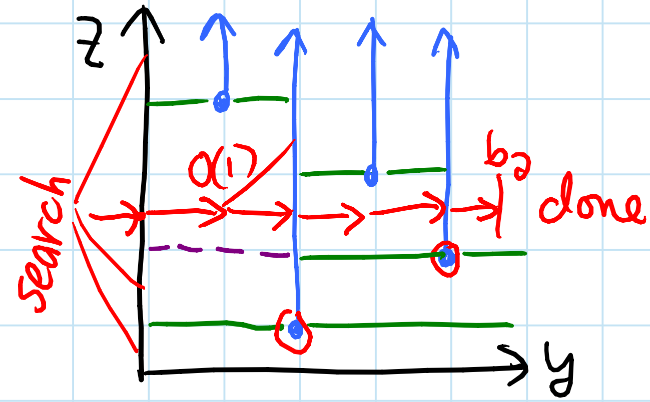
    $[a_1, b_1] \times (a_2, \infty) \times (-\infty, b_3)$ in left(v')

    $\cup$ $[a_1, b_1] \times (-\infty, b_2) \times (-\infty, b_3)$ in right(v')
  - each subtree stores:
- range tree on x:
  - query: $[a_1, b_1] \times (-\infty, b_2) \times (-\infty, b_3)$ & symmetrics
  - reduces to $O(\lg n)$ subtrees $+O(\lg n)$

    $(-\infty, \infty) \times (-\infty, b_2) \times (-\infty, b_3)$ (2D problem)
  - each subtree stores:

- 2D "hive" DS searches for b_3 in z order & then outputs each match in $O(1)$ time until reaching $y = b_2$ (& symmetrics)



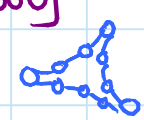


- $O(\lg n)$ total searches for b_3 (& a_3) in lists of length $O(n)$
- fractional cascading $\Rightarrow$ $O(\lg n)$ time total

+ $O(\lg n)$ for "range tree" in y

+ $O(\lg n)$ for "range tree" in z

Kinetic survey: [Guibas – DS Handbook 2005] $\boxed{\text{L4 p8}}$

- 2D convex hull [Basch, Guibas, Hershberger 1999]
  - also diameter, width, min. area/perim. rectangle
  - efficiency = $O(n^{2+\varepsilon})/\Omega(n^2)$
  - $\boxed{\text{OPEN}}$: 3D?
- $(1+\varepsilon)$-approximate diameter, smallest disk/rectangle in $(1/\varepsilon)^{O(1)}$ events [Agarwal & Har-Peled – SODA 2001]
- smallest enclosing disk: [Demaine, Eisenstat, Guibas, Schulz – FWCG 2010] efficiency $O(n^{3+\varepsilon})/\Omega(n^2)$
- Delaunay triangulation [Albers, Guibas, Mitchell, Roos – IJCGA 1998]
  - $O(1)$ efficiency
  - $\boxed{\text{OPEN}}$: how many changes? $O(n^{2+\varepsilon})$ & $\Omega(n^2)$ [Rubin – FOCS 2013]
- any triangulation:
  - $\Omega(n^2)$ changes even with Steiner points [Agarwal, Basch, de Berg, Guibas, Hershberger – SoCG 1999]
  - $O(n^{2+1/3})$ events [Agarwal, Basch, Guibas, Hershberger, Zhang – WAFR 2000]
  - $\boxed{\text{OPEN}}$: $O(n^2)$?
  - $O(n^2)$ events for pseudotriangulations 
- collision detection [Kirkpatrick, Snoeyink, Speckmann 2000] [Agarwal, Basch, Guibas, Hershberger, Zhang 2000] [Guibas, Xie, Zhang 2001] ← 3D
- MST
  - $O(m^2)$ easy: $\boxed{\text{OPEN}}$: $o(m^2)$? ↗ sorted order of edge weights
  - $O(n^{2-1/6})$ for H-minor-free graphs (e.g. planar) [Agarwal, Eppstein, Guibas, Henzinger – FOCS 1998]

Problem 1: Design a kinetic priority queue DS maintaining min on n affinely moving points using $O(n \frac{\lg n}{\lg \lg n})$ total events,
and $O(\lg n)$ time per event
and $O(n)$ space
(no insert/delete)

[da Fonseca & de Figueiredo 2002]

Problem 2: Show that the kinetic heap [L04] can incur $\Omega(n \lg n)$ events in the worst case
[and/or show that your DS for Problem 1 can incur $\Omega(n \lg n / \lg \lg n)$ events in the worst case]