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

```
1 1 1 1 1 1 1
1 1 1 1 1 1 1
```
```
1 1 1 1 1 1 1
1 1 1 1 1 1 1
```

<these nodes need recursive ptr. changes>
<this node needs direct change (no recursion)>

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

L₁: 1, 2, 3, 4, 13
L₂: 0, 5, 7, 9, 11
L₃: 6, 8, 10, 12, 14

search(x):
- binary search in L₁ for pred/succ. of x
- go to prev & next down pointer
- walk left/right 0 or 1 steps to predecessor/successor of x in Lᵢ
- pred./succ. in Lᵢ via original item ptrs.
- repeat (i += 1) from pred/succ. in Lᵢ (not Lᵢ)

e.g. x = 5, 9, 12
3D range queries in $O(lg n + k)$: top-down review
- static
- reporting only: $k = \# \text{ desired pts. in box}$

- "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 $lca(a_3, b_3) = v$ + $O(lg n)$
  - reduce to 2 queries at node $v$:
    $[a_1, b_1] \times [a_2, b_2] \times (a_3, \infty)$ in $left(v)$
    $v \times [a_1, b_1] \times [a_2, b_2] \times (-\infty, b_3)$ in $right(v)$
  - 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 $v'$:
    $[a_1, b_1] \times (a_2, \infty) \times (-\infty, b_3)$ in $left(v')$
    $v \times [a_1, b_1] \times (-\infty, b_2) \times (-\infty, b_3)$ in $right(v')$
  - each subtree stores:

- range tree on $x$:
  - query: $[a_1, b_1] \times (-\infty, b_2) \times (-\infty, b_3)$ & symmetric
  - reduces to $O(lg n)$ subtrees + $O(lg n)$
  - each subtree stores: $(-\infty, \infty) \times (-\infty, b_2) \times (-\infty, b_3)$ (2D problem)
- 2D "hive" DS searches for $b_3$ in $\mathcal{O}(\log n)$ order & then outputs each match in $\mathcal{O}(1)$ time until reaching $y = b_3$ (\& symmetrics)

- $O(\log n)$ total searches for $b_3$ (\& $a_3$) in lists of length $O(n)$
- Fractional cascading $\Rightarrow O(\log n)$ time total
  + $O(\log n)$ for "range tree" in $y$
  + $O(\log n)$ for "range tree" in $z$
Kinetic survey:

- 2D convex hull
  - also diameter, width, min. area/perm. rectangle
  - efficiency $\Omega(n^{2+\varepsilon})/\Omega(n^2)$
  - **OPEN**: 3D?

- $(1+\varepsilon)$-approximate diameter, smallest disk/rectangle in $(1/\varepsilon)O(1)$ events
  - smallest enclosing disk: $\Omega(n^{3+\varepsilon})/\Omega(n^2)$
  - Delaunay triangulation
    - $O(1)$ efficiency
    - **OPEN**: how many changes? $\Omega(n^{3+\varepsilon}) & \Omega(n^2)$

- any triangulation:
  - $\Omega(n^2)$ changes even with Steiner points
    - $O(n^{2+\varepsilon})$ events
    - **OPEN**: $O(n^2)$?
  - $O(n^2)$ events for pseudo triangulations

- collision detection
  - [Kirkpatrick, Snoeyink, Speckmann 2000]
  - [Agarwal, Basch, Guibas, Hershberger, Zhang 2000]
  - [Guibas, Xie, Zhang 2001] \xRightarrow{3D}

- MST
  - sorted order of edge weights
  - $O(m^2)$ easy; **OPEN**: $o(m^2)$?
  - $O(n^{2-1/6})$ 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 \( O(n \frac{\log n}{\log \log n}) \) total events, and \( O(\log n) \) time per event and \( O(n) \) space (no insert/delete) \[\text{[da Fonseca & de Figueiredo 2002]}\]

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