Skip Lists

William Pugh (1989)

- Easy to implement (as compared to balanced trees)
- Maintains a dynamic set of n elements in $O(\log n)$ time per operation in expectation and with high probability (w.h.p.)

One Linked List

One (Sorted) linked list

$\boxed{14} \leftrightarrow \boxed{23} \leftrightarrow \boxed{34} \leftrightarrow \boxed{42} \leftrightarrow \boxed{50} \leftrightarrow \boxed{59} \leftrightarrow \boxed{66} \leftrightarrow \boxed{72} \leftrightarrow \boxed{79}$

Searches take $\Theta(n)$ time in worst case

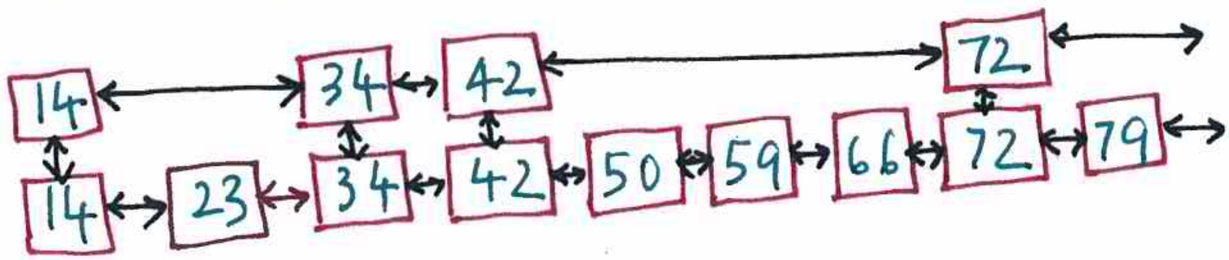
Suppose we had two sorted linked lists

- each element can appear in one or both lists

Two Linked Lists

Express and local subway lines
(à la New York City 7th Avenue Line)

- Express line connects a few of the stations
- Local line connects all stations
- Links between lines at common stations



Searching in Two Linked Lists

Search(x):

- Walk right in top linked list (L_1) until going right would go too far
- Walk down to bottom linked list (L_2)
- Walk right in L_2 until element found (or not)

Search(59)

Analysis

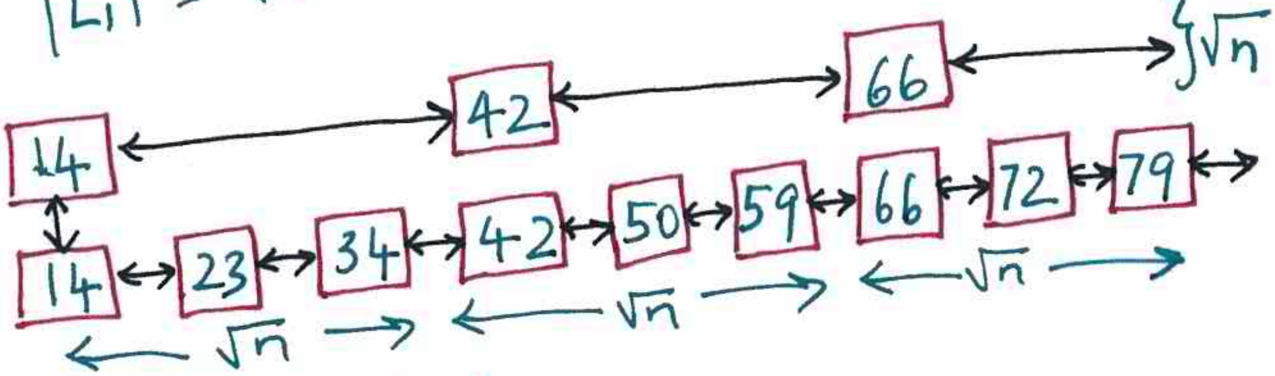
Search cost $\approx |L_1| + \dfrac{|L_2|}{|L_1|}$

Minimized when terms are equal

$|L_1|^2 = |L_2| = n$

$|L_1| = \sqrt{n}$

Search is $\Theta(\sqrt{n})$



More Linked Lists

2 sorted lists $\Rightarrow$ $2 \cdot \sqrt{n}$

3 sorted lists $\Rightarrow$ $3 \cdot \sqrt[3]{n}$

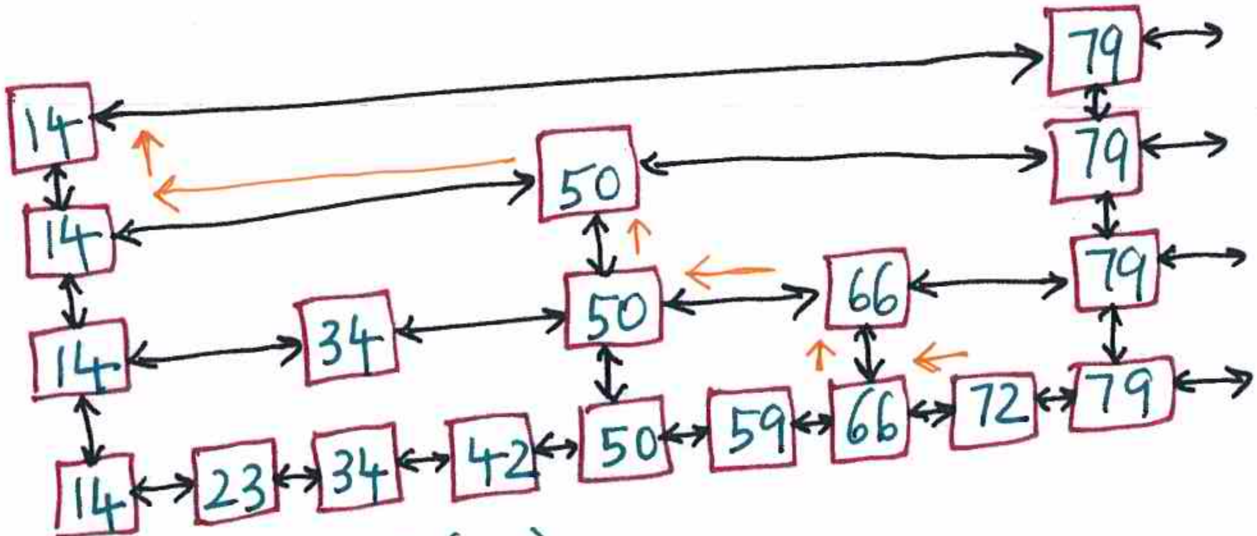
k sorted lists $\Rightarrow$ $k \cdot \sqrt[k]{n}$

lg n sorted lists $\Rightarrow$ $\lg n \cdot \sqrt[\lg n]{n}$

$= 2 \lg n$

like a binary tree!

Searching in lg n Linked Lists



Try search(72)

Insert (x)

To insert an element x into a skip list

- Search(x) to see where x fits into bottom list
- Always insert into bottom list
- Insert into some of the lists above
  which ones?
- Flip fair coin
  if HEADS : promote x to next level up
  else stop

$\underbrace{\text{next level up}}$ this may be newly created

WHY ARE SKIP LISTS GOOD?

Warmup Lemma: # levels in n-element skip list is $\underbrace{O(\lg n)}_{c \cdot \lg n}$ w.h.p. $\rightarrow$ prob $1 - \frac{1}{n^\alpha}$

related

Proof: Failure probability (not $\leq c \lg n$ levels)

$$= \Pr\{> c \lg n \text{ levels}\}$$

$$= \Pr\{\text{some element got promoted} > c \lg n \text{ times}\}$$

$$\leq n \cdot \Pr\{\text{element } x \text{ got promoted} > c \lg n \text{ times}\}$$ by union bound

$$= n \cdot \left(\frac{1}{2}\right)^{c \lg n}$$

$$= \frac{n}{n^c}$$

$$= \frac{1}{n^{c-1}} = \frac{1}{n^\alpha} \qquad \alpha = c-1$$

SEARCH

Theorem: Any search in an n-element skip list costs $O(\lg n)$ w.h.p.

Cool idea: Analyze search backwards

- search starts [ends] at node in bottom list.
- At each node visited:
  - If node wasn't promoted higher (tails here) then we go [came from] left
  - If node was promoted higher (heads here) then we go [came from] up
- Stop [start] when we reach top level or $-\infty$

Look at ← ↑ arrows on page 4

Proof of Theorem

- Search (backwards) makes "up" and "left" moves each with probability $1/2$
- Number of moves going "up" $<$ # levels $\leq c \cdot \lg n$ w.h.p. (by Warmup Lemma)

- Total number of moves = number of coin flips until you get $c \lg n$ heads ("up" moves)

Claim: Number of coin flips until $c \lg n$ heads $= O(\lg n)$ w.h.p.

CHERNOFF BOUNDS

Theorem: Let Y be a random variable representing the total number of $\frac{\text{heads}}{\text{Tails}}$ in a series of m independent coin flips, where each flip has a probability p of coming up $\frac{\text{heads}}{\text{Tails}}$.

Then for all $r > 0$, we have

$$\Pr[Y \geq E[Y] + r] \leq e^{-\frac{2r^2}{m}}$$

Lemma: For any c, there is a constant d such that with high probability (w.h.p.) the number of heads in flipping $d \lg n$ fair coins is at least $c \cdot \lg n$. ← This is our claim from page 7!

Proof: Let Y be the number of tails when flipping fair coin $d \lg n$ times. $p = 1/2$

$$m = d \lg n, \text{ so } E[Y] = \frac{1}{2} m = \frac{d \lg n}{2}$$

We want to bound the probability of fewer than $\leq c \cdot \lg n$ heads $=$ the probability of getting at least $\geq d \cdot \lg n - c \lg n$ tails.

Proof of Lemma (contd.)

$$\Pr\left[Y \geq (d-c)\lg n\right] = \Pr\left[E[Y] + \underbrace{\left(\tfrac{d}{2}-c\right)\lg n}_{r}\right]$$

Choose $d = 8c \Rightarrow r = 3c\lg n$

By Chernoff, prob of $\leq c \cdot \lg n$ heads

$$\leq e^{-\frac{2r^2}{m}}$$

$$= e^{-\frac{2(3c\lg n)^2}{8c\lg n}}$$

$$= e^{-\frac{9}{4}c\lg n}$$

$$\leq e^{-c\lg n}$$

$$\leq 2^{-c\lg n} \qquad (e > 2)$$

$$= \frac{1}{2^{c\lg n}}$$

$$= \frac{1}{n^c}$$

☺ for Lemma

Proof of theorem (finally!)

event A : number of levels $\leq c \lg n$ w.h.p.

event B : number of moves until $c \cdot \lg n$ "up" moves $\leq d \lg n$ w.h.p.

event A and event B are not independent

Want to show $\Pr(\underbrace{\text{event A \& event B}}_{\text{w.h.p.}})$ high

$$\Pr\left(\overline{\text{event A \& event B}}\right) = \Pr\left(\overline{\text{event A}} + \overline{\text{event B}}\right)$$

$$\leq \Pr\left(\overline{\text{event A}}\right) + \Pr\left(\overline{\text{event B}}\right) \quad \text{(union bound)}$$

$$\leq \frac{1}{n^{c-1}} + \frac{1}{n^c}$$

$$= O\left(\frac{1}{n^{c-1}}\right)$$

$\therefore$ $\Pr(\text{event A \& event B})$ w.h.p.

Search in $O(\lg n)$ w.h.p.

 for theorem.