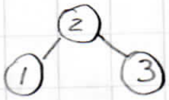
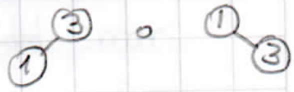
Si inserto 3 2 1 quizás el orden de cosas que pasan es distinto pero obtengo la misma distribución

¿Y cómo borramos? tenemos que imaginarnos cómo serían los posibles árboles a los cuales nunca insertamos el elto que queremos borrar. Si queremos borrar una hoja, la borramos y listo. Pero un nodo interno?

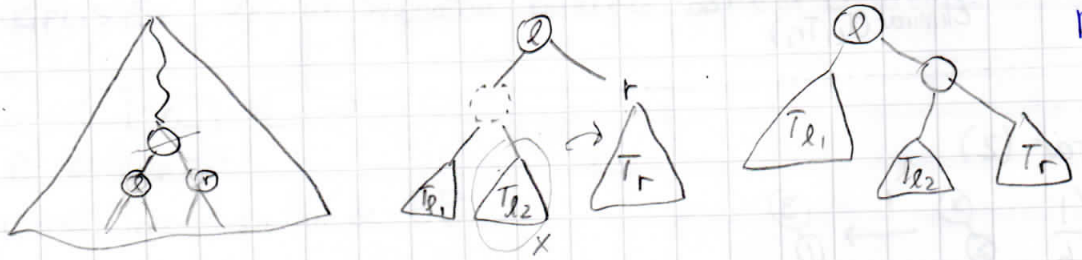


→ si saco el 2 : si nunca hubiese insertado 2 yo podría haber tenido



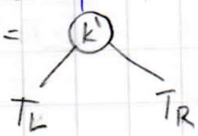
No tenemos suficiente información !! quién fue primero?

→ simulamos con random



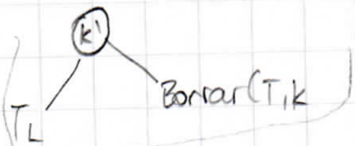
Borrar (T, k) (borrar busca, eliminar "parte" con nodo vacío y retorna árbol donde puso algo en ese nodo vacío)

Sea T =



si $k < k'$ retornar

Si $k > k'$ retornar

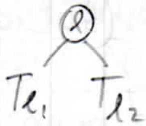


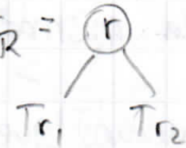
Si $|T| = 1$ retornar □

retornar Eliminar (T_L, T_R)

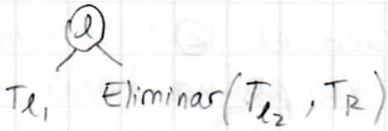
devuelve árbol que produce raíz $+ T_L + T_R$

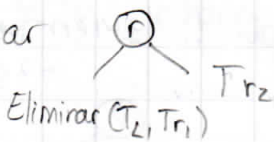
Eliminar(T_L, T_R)

Sea $T_L =$ 

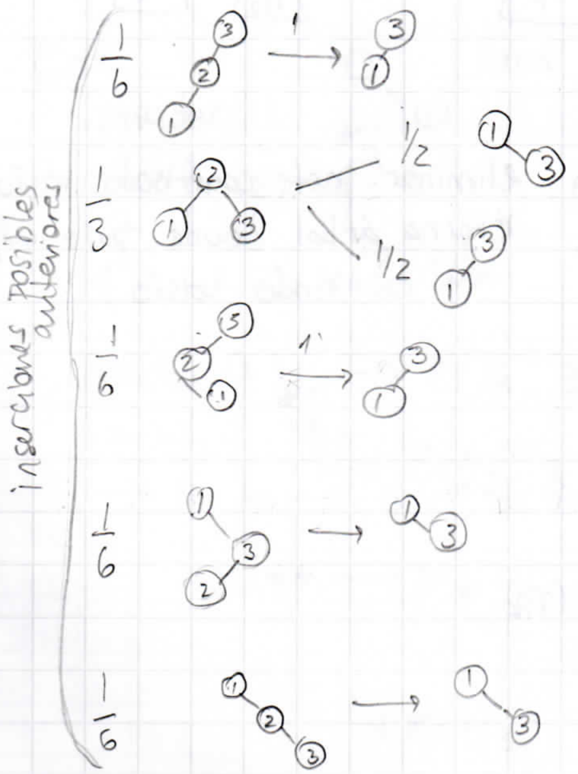
, $T_R =$ 

si random$[0,1) < \dfrac{|T_L|}{|T_L| + |T_R|}$

retornar 

retornar 

Borar (2)



Aux # 10

2015年11月 23日 (月)

Algoritmos aleatorizados → tira monedas
y probabilísticos → hay $\mathbb{P}$ asociado al tiempo / correctitud.

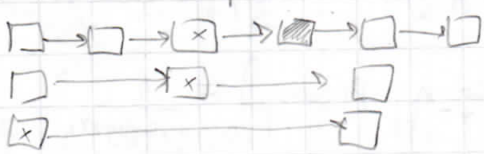
Algoritmos tipo:

- MonteCarlo: se detienen siempre pero pueden equivocarse → si el algoritmo tiene 2 respuestas (sí / no)
  - one-sided error
  - two-sided error.
- Las Vegas: siempre es correcto pero pueden no detenerse

$\boxed{\text{P1}}$ Skiplists: "aplicar búsqueda binaria a lista enlazada"

- idea 0: ordenar
- Queremos los "saltos" de BB

- idea: metro expreso



L_1, $|L_1| = l_1 = n$
L_2
L_3 $|L_i| = l_i$
⋮

Si hay 2 niveles, buscar cuesta:

$\leq l_2 + \frac{l_1}{l_2} = l_2 + \frac{n}{l_2}$ ⇒ minimizo c/r a l_2
("¿Cuántos eltos. pongo en l_2?")

$\Rightarrow C_{min} = 2\sqrt{n}$

Si tenemos cualquier cantidad de niveles, lo ideal sería simular un árbol binario ⟹ links de largo 1, 2, 4, ...

¿Cómo logro que se vea así si hay inserciones?
Idea desde árboles balanceados: cada piso tien 1/2 de los nodos del piso de abajo

⟹ insertar(x)
- Buscar en L_1 la posición que le corresponde a x
- Tiro una moneda hasta que salga sello
- Si tiré la moneda K veces, x estará en las listas $L_1, L_2, \ldots L_k$

$\mathbb{P}$[un elem. esté en la lista K] $= \frac{1}{2^{(K-1)}}$ → siempre tiro la moneda 1 vez.

Hay que analizar:
- Tamaño de la estructura
- Tiempo de ejecución

1) Con alta probabilidad, si hay n elementos hay $\Theta(\log n)$ pisos.
¿Por qué?

$\mathbb{P}[x \text{ esté en más de } c \cdot \log_2 n \text{ pisos}] = \frac{1}{2^{c \cdot \log_2 n}}$

$= \frac{1}{n^c}$, pero esto es para 1 x!

Queremos acotar la $\mathbb{P}$. de que cualquiera sea más alto que $c \log_2 m$

$\mathbb{P}[\text{cualquier } x, \ldots] = \mathbb{P}[p(x_1) \cup p(x_2) \cup \ldots p(x_n)]$

$$\leq \sum_i \mathbb{P}[p(x_i)] = \frac{n}{n^c} = \frac{1}{n^{c-1}}$$

el número de pisos es $> c \cdot \log_2 n$ con prob $\leq \frac{1}{n^{c-1}}$

Otra forma es ver la esperanza.

2) ¿cuántos nodos tiene la estructura? (en promedio)

- Nos gustaría saber $\mathbb{E}(l_i)$

$$E(l_i) = \sum_{x=0}^{\infty} x \, \mathbb{P}[l_i = x] = ?$$

VARS INDICADORAS: v.a con valor 0 ó 1

Definimos X_{ij} = "el elemento $x_i \in L_j$"

$\Rightarrow \mathbb{E}(l_j) = E\left(\sum_{i=1}^{n} X_{ij}\right)$

$$= E\left(\sum_i^n \frac{1}{2^{j-1}}\right) = \frac{n}{2^{j-1}}$$

$$E(n^\circ \text{ nodos}) = E\left(\sum_{j=1}^{\infty} l_j\right)$$

$$= \sum_{j=1}^{\infty} \frac{n}{2^{j-1}} = \sum_{i=0}^{\infty} \frac{n}{2^i} = 2n$$

$E(n^\circ \text{ de pisos}) = ?$ $\quad H_i = \begin{cases} 0 & \text{si } L_i \text{ vacía} \\ 1 & \text{si no} \end{cases}$

$\Rightarrow \mathbb{E}(n^\circ \text{ de pisos}) = \mathbb{E}\left(\sum_i H_i\right)$

- $H_i \leq l_i \Rightarrow \mathbb{E}(H_i) \leq \frac{n}{2^{i-1}}$, $\quad H_i \leq 1$

$$\mathbb{E}\left(\sum H_i\right) = \sum_{i=1}^{\infty} H_i = \sum_{i=1}^{\lfloor \log n \rfloor} H_i + \sum_{i=\lfloor \log n \rfloor + 1}^{\infty} H_i$$

$$\leq \sum_{i=1}^{\lfloor \log n \rfloor} 1 + \sum_{i=\lfloor \log n \rfloor + 1}^{\infty} \frac{n}{2^{i-1}} \leq \log n + \sum_{i=\lfloor \log n \rfloor + 1}^{\infty} \frac{n}{2^k} \cdot \frac{1}{2^i} = \log n + \sum_{i=0}^{\infty} \left(\frac{n}{2^k}\right) \cdot \frac{1}{2^i}$$

$\geq \lfloor \log n \rfloor + 1$