

Dictionaries

Note Title

10/1/2009

DI. Data Structures Abstracts

Insert (k, x)

Find (k)

Delete(k)

Find Next(k) (if ordered)

② Local Dictionaries Concen

- Arevalo Choleno

- File more to front

- " transpose

- Arboles hinaños

" " deatorigados

- Ship Lists ✓

- AVL

- Arboles 2-3

- " Red Black

[B-arboles
- van Emde Boas Arboles

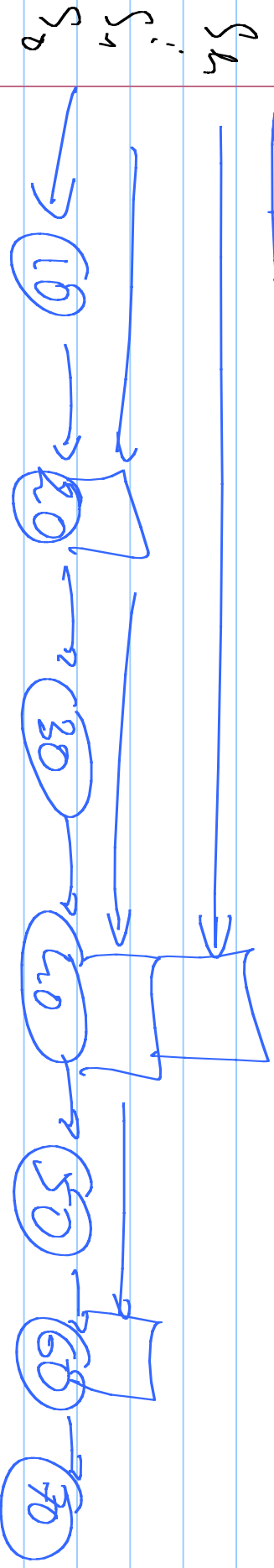
- Splay Arboles

③ Skip List

Preguntas:

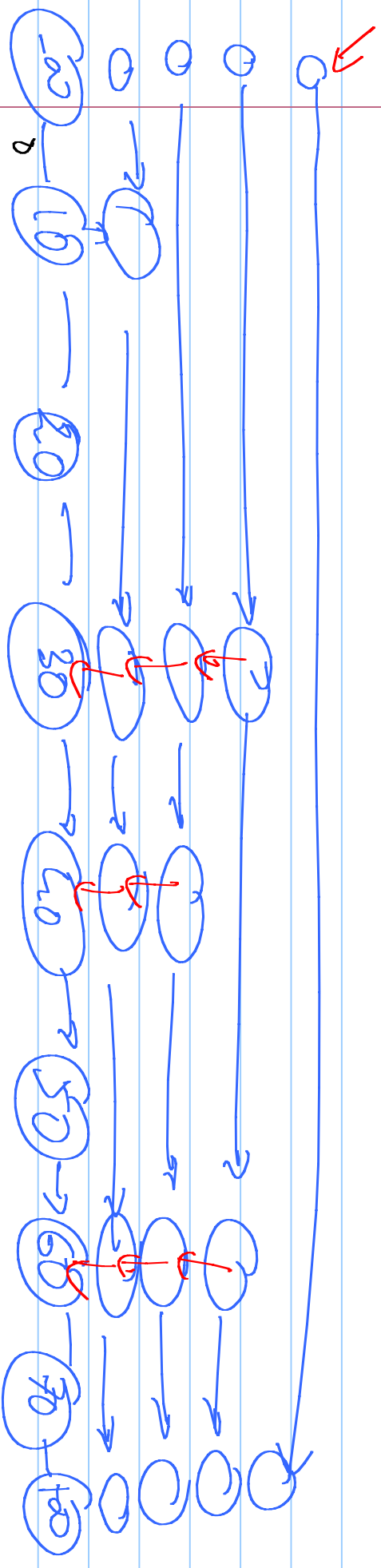
a) Como hacer una busqueda binaria en una lista?

⇒ Resumenes de Resumenes



Espace "altura" = cantidad de resumenes = $\lg n$
Punteros $n + \frac{n}{2} + \dots + \frac{n}{2^k} \leq \epsilon n$

b) Como utilizar el agar para las infecciones?
→ los elementos son elegidos "al agar" para estar en cada resumen



③ Operaciones

Search(k)

Inicio a la "í" / "j" = primer elemento
de la lista mas corta,
el resumen de nivel mas
alto)

- Hasta que estemos en S_0

* Bajar de un nivel

* Buscar (Secuencialmente) en S_i para el
intervalo que contiene k .

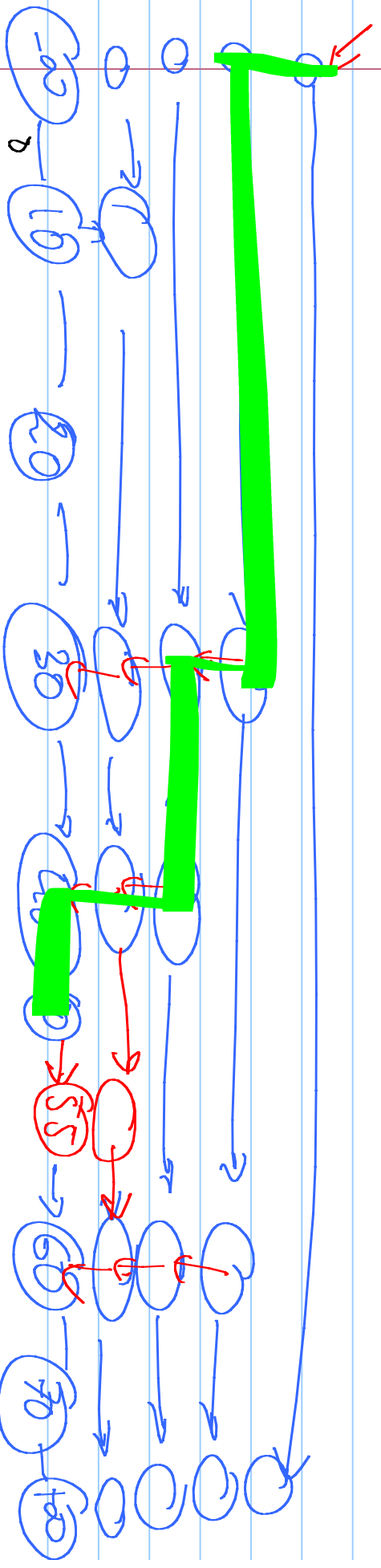
Insert (k, x)

^{En} Search (k), memorizando los punteros cada vez que haya

• Lee una "torre" para (k, x),
cada nivel adicional con probabilidad $p = \frac{1}{2}$

• Configa los punteros $\rightarrow$ "necesarios"

Ejemplo: Insert(55) con secuencia aleatoria (10)

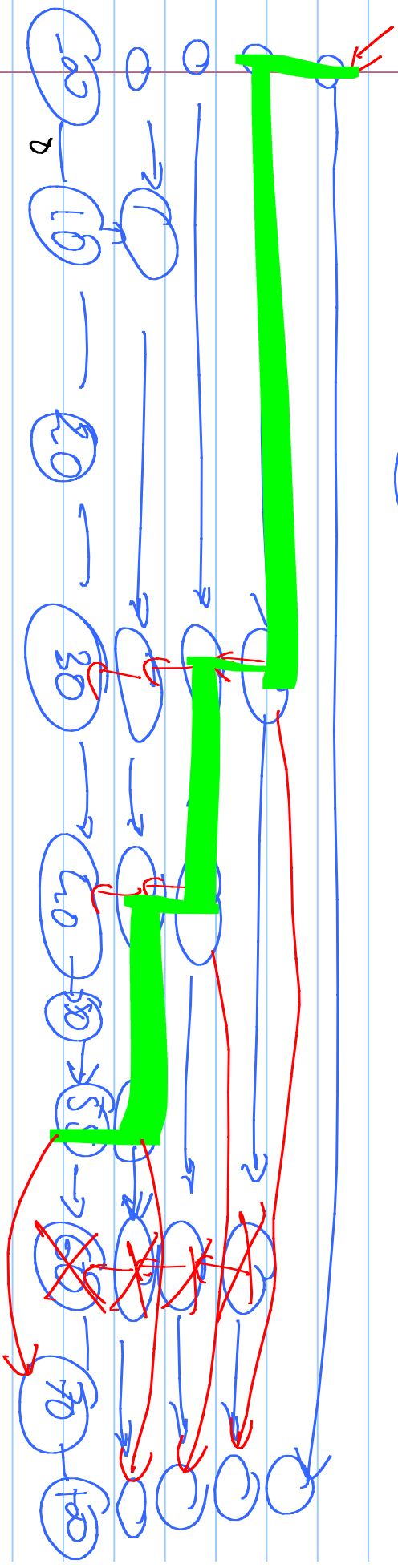


Delete (k)

Search (k), recordando (---)

Remueve k de cada lista, corrigiendo los punteros
Remueve el nivel mas alto si vacillo.

Ejemplo Delete (60)



⑤ Análisis / Discussion

- Fácil a implementar

- En teoría, tiempo $O(\lg n)$ } en promedio,
espacio 2^n

- En práctica, No es tan bueno

- Instancias "malas" estan estructuradas,
que se puede usar en otros algoritmos.

- "Predictive Branching" (efecto ^{positivo} es destruido
por el azar, (que es malo)).

