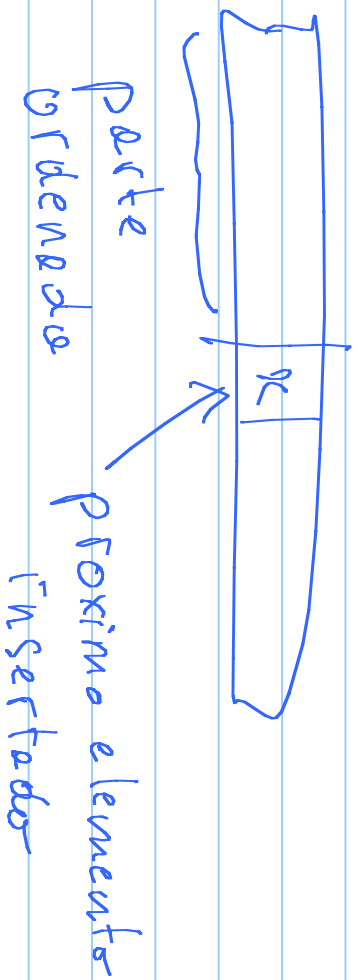


Comparacion de Medidas de Dificultades

II Un otro algoritmo adaptivo de ordenamiento

II. Straight Insertion Sort

Una regla (ordenado)



	Complejidad Seq	BS
Mejor	$O(n)$	$O(n \log n)$
Peor	$O(n^2)$	$O(n^2)$

I.2 Local Insertion Sort

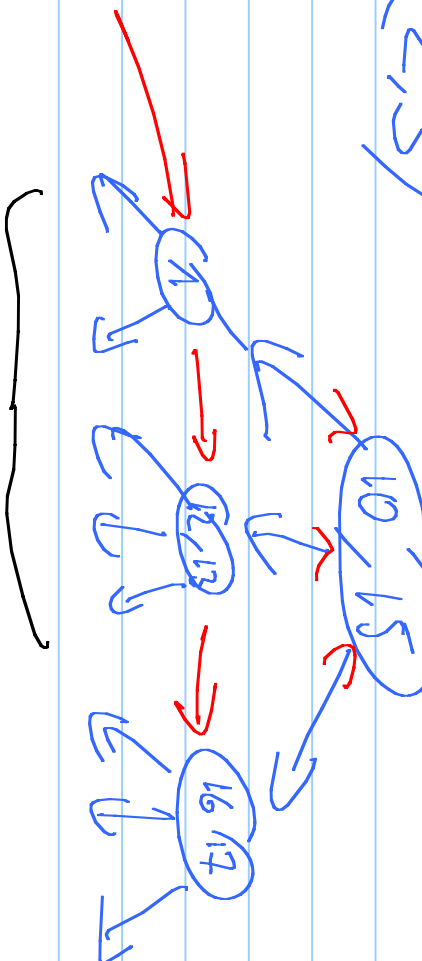
"Finger Search Tree" e.g. Splay Tree (may be simple)

L_2 Arbor $(2,3)$

Algs

$\text{pos} := \text{root}$

$\text{Insert}(A[i])$



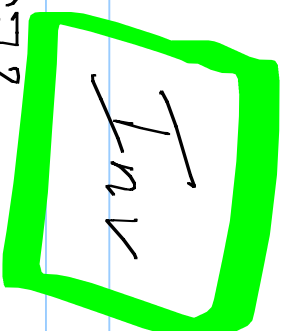
Recesses $\log$
 $O(\log n)$ or $O(1)$

$\text{pos} := \text{root}$

$\text{Insert}(A[i])$

$\text{pos} := \text{Extract}(\text{Min})$ $\text{pos} := \text{Extract}(\text{Max})$

1.3 Medida de Dificultad



$$P_i = \# \{ j, j < i : a_j > a_i \}$$

$$Inv = \sum_{i=1}^n \sum_{j=1}^{i-1} \mathbb{1}_{\{a_j > a_i\}} = \sum P_i$$

Complejidad de **Straight Insertion Sort**? $O(n + Inv)$

$$\sum P_i = Inv \in [0, n^2]$$

Complejidad de **Local Insertion Sort**?

$$\sum \log P_i \leq n \log \sum \frac{P_i}{n} = n \log \left(\frac{Inv}{n} \right)$$

II Cual algoritmo es el mejor?

II.1 Los algoritmos que ya vimos:

- a. Merge Sort para "ascending runs" $n + n \lg R_{\text{runs}}$
- b. Merge Sort para "shuffled ascending subsequences" $n \lg S_{\text{US}} + n \lg S_{\text{US}}$
- c. Local Insertion Sort $n \lg \frac{I_{\text{inv}}}{n}$

Cual es el mejor?

IV.2 Reduction

III.3 Independence

II Mapa Pactual
III.1 Otras Medidas

III. 2 ¿Cual relaciones conocemos? ¿Adivinamos?

III.3 Mapa Completa Recia

