

Algoritmos de Ordenamiento

¿Todos son en $O(n \log n)$?

NO! • "Insertion Sort" $O(n^2)$

• Counting Sort $\in O(n \log n)$

$O(n)$ diferentes
"pocos" números

Usa de dominios discretos y finitos

↳ Link con "hashing"

Q1 Counting Sort ($n, \sigma, A \in [\sigma]^n$) $O(n + \sigma)$

1. For $j = 1$ to σ do $C[j] \leftarrow 0$ $O(\sigma)$

$\hookrightarrow [\sigma] = \{1, \dots, \sigma\}$

2. For $i = 1$ to n do $C[A[i]]++$ $O(n)$

3. $p \leftarrow 1$ // position en A $O(1)$

4. For $j = 1$ to σ do
for $k = 1$ to $C[j]$ $O(n + \sigma)$

$A[p+j] \leftarrow j$

Espace $O(\sigma)$

Bucket Sort

$O(n+s)$

1. For $j \in 1$ to s do $C[j] \leftarrow 0$
2. for $i = 1$ to n do $C[A[i]]++$

3 $P[1] \leftarrow 1$

4. for $j \in 2$ to s do

$P[j] \leftarrow P[j-1] + C[j-1]$

Note
que Bucket sort

5. for $i \in 1$ to n

$B[P[A[i]]++] \leftarrow A[i]$

$O(s+n)$

es estable:
los elementos con valores
iguales conservan su orden

③ Radix Sort

Sea N el tamaño de A
 $\leq$ el tamaño del dominio

$$S_i: n \leq b$$

$\rightarrow$ Bucket Sort (BS) puede ordenar

$$S_i: b = n^2$$

$\rightarrow$ BS Simple $\text{va ordenar en } O(S) = O(n^2)$

$\rightarrow$ BS puede ordenar en $O(n)$

- 1 ves con $\lg n$ bits en $O(n)$

- 1 ves con $\lg n$ bits en $O(n)$

$$S_i: b = n^c$$

$\rightarrow$ BS puede ordenar en $O(c n)$

c veces con $\lg n$ bits en $O(n)$

Complexidad (de Radix Sort)

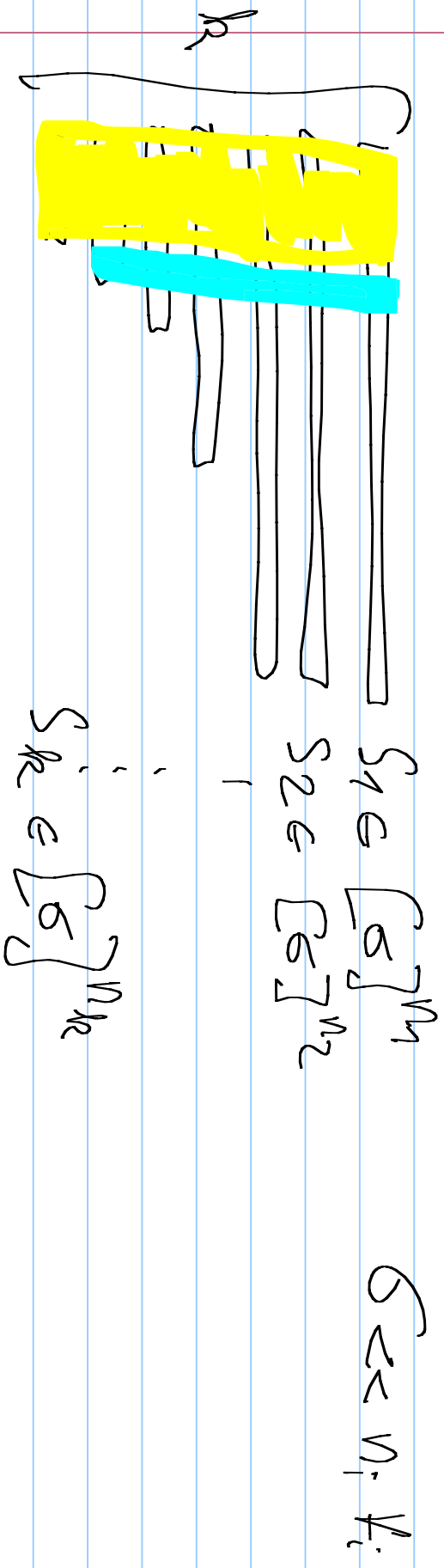
• en espacio: Si $S = n^c$, $3nc O(n)$

En práctica, se puede usar $\frac{\lg n}{2}$ bits por cada etapa
 $\Rightarrow \sqrt{n}$ casilleros en C , tiempo $2\sqrt{n} c O(n)$
• en tiempo

$$O\left(n \left\lceil \frac{\lg S}{\lg n} \right\rceil\right) \quad \text{para } S \text{ general}$$
$$\left(n^{c-1} < S \leq n \Rightarrow c = \left\lceil \frac{\lg S}{\lg n} \right\rceil\right)$$

Q String Sort

Pb: Ordena k strings sobre $[c]$
de longo total $n = \sum n_i$



Como ordenarlas?

- ① Superenganos $n_1 \geq \dots \geq n_k$
- ② Ordenar con n_k símbolos
 $\hookrightarrow$ Ordenar con 1 símbolo $O(k)$ $O(k n_k)$
- ③ Ordenar en cada block separadamente
 $\hookrightarrow$ definidos para su prefijo de tamaño n_k

Complejidad total
tiempo $O(n)$ ($\equiv$ la superficie)
espacio $O(k)$

