

Ordenamiento

Note Title

10/5/2010

n elementos en un dominio de tamaño $\leq k$
 n_a elementos a para a en el dominio.

* Counting Sort

Input: n números en desorden
dentro de $[1..k] = \{1, \dots, k\}$

Output: n números e

Algo $\left[\begin{array}{l} \text{Crear arreglo } C[1..k] = [0, \dots, 0] \\ \text{Para cada Valor } A[i] \end{array} \right.$

Complejidad

② Bucket Sort

Input n objetos, con llaves $A[i] \in [1..c]$
 $\forall i \in [1..n]$

Output n objetos ordenados de manera estable ($\equiv$ el orden final entre x y y si $A[x] = A[y]$ es igual que en el input)

Algoritmo

$O(n+c)$

- Iniciar c "buckets" al vacío $B[1..c]$
(= baldes)
- Para $i \in [1..n]$ agregar el objeto i
en $B[A[i]]$
- Iniciar Output O
- Para $a \in [1..c]$, vaciar $B[a]$ en O

③ "Radix Sort"

Ejemplo: años de nacimiento

A = 1987, 1988, 1989, 1989, 1989, 1986, 1987, 1986, 1980, 1982, 1988, 1976

	0	1	2	3	4	5	6	7	8	9
B ₁	1980		1982				1986 1986 1976	1987 1987	1988 1988	1989 1989 1989

no funciona

~~1980, 1982, 1976, 1986, 1986, 1987, 1987, 1988, 1988, 1989, 1989, 1989~~

$A_n =$ 1980 1982 1986 1986 1976 1987 1987 1988 1988 1989 1989 1989
 Stable Bucket Sort

Más General
 $b = 2$ Buckets,

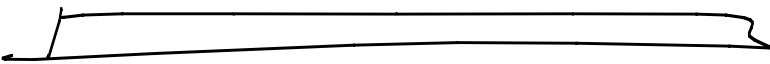
$\Rightarrow$ Complejidad ~~$O(n \log 5)$~~

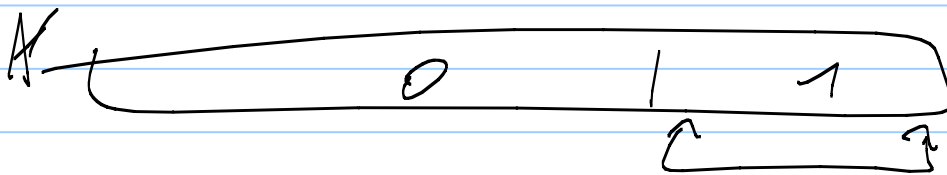
Cada Bucket Sort cuesta $O(n)$
 * $\lg b$ etapas

$O(n \log 5)$

$b > 2$ pero $\leq \sqrt{n}$ $\rightarrow$ La dificultad es el espacio de bucket Sort.

Remarks $G=2$, bucket sort en espacio n :

A 



④ Volviendo al Modelo de Comparaciones

Se puede proveer de las repeticiones
SIN tener acceso a los valores?
(i.e. en el modelo de comparaciones)

Pb Si Pocos valores α dentro de $[1..n]$
son utilizadas,

Solucion : mantener un diccionario de tamaño α
el tamaño práctico de la instancia
 $\rightarrow O(n \lg \alpha) \subset O(n \lg n)$

$\approx \min\{\alpha, n\}$

Bonus

* Insertion Sort

$$n \lg n - \sum_{a=1}^{\sigma} n_a \lg n_a = H(n_1, \dots, n_{\sigma})$$

* Merge Sort

$$n \lg n - \sum n_a \lg n_a = H(n_1, \dots, n_{\sigma})$$