

Counting sort

giovedì 2 giugno 2022

11:57

- Algoritmo ordinamento non basato sui confronti
- $O(n) \rightarrow O(n+k)$, k = range numeri
- Range da ordinare sia $1 \dots k$ con $k = O(n)$
- Algoritmo non in loco

$A = [1 \dots n]$

$B = [1 \dots n] \rightarrow$ Informazioni aggiuntive di A

$C = [1 \dots k] \rightarrow$ Contatore dei valori

- Azzera C
- Conto quante volte compare ogni elemento di A Memorizzandolo in C
- Somma in C gli elementi da sinistra verso destra
- Scorri A dalla fine verso l'inizio e piazza i valori a seconda di C

1) $A = 4, 1, 3, 1, 4, 4$
 $C = 0, 0, 0, 0$

2) $C = 2, 0, 1, 3$

3) $C = 2, 2, 3, 6$

4) $A[6] = 4$
 $B[C[4]] = 4$
 $C = 2, 2, 3, 5$

$A[5] = 4$
 $B[C[4]] = 4$
 $C = 2, 2, 3, 4$

$A[4] = 1$
 $B[C[1]] = 1$
 $C = 1, 2, 3, 4$

CountingSort(A[], B[], C[], k):

1

For $i = 1$ to k : $2k$
 $C[i] = 0$

2

For $i = 1$ to $\text{len}(A)$: $3n$
 $\text{Pos} = A[i]$
 $C[\text{pos}]++$

3

For $i = 2$ to k : $2k$
 $C[i] = C[i] + C[i-1]$

4

For $i = \text{len}(A)$ downto 1 : $5n$
 $p1 = A[i]$
 $p2 = C[p1]$
 $B[p2] = A[i]$
 $C[p1]--$

$$T(n) = 4k + 8n = \theta(n + k)$$