Esercitazione 5 (7 aprile '09)

Dott.ssa Gaia Maselli (maselli@di.uniroma1.it)

Problema 4.12

Dato l'algoritmo:

Algoritmo Algo (array A, interi l,r)

1. If (l ≥ r) then return
2. If (A[l] > A[r]) then scambia A[l] ed A[r]
3. Algo(A, l+1, r-1)
4. If (A[l] > A[l+1]) then scambia A[l] e A[l+1]
5. Algo(A, l+1, r)

1) Scrivere una relazione di ricorrenza che descriva il numero di volte che due elementi dell'array A vengono confrontati, in funzione della lunghezza dell'array n = r-l+1
2) Trovare il tempo di esecuzione dell'algoritmo Algo risolvendo la relazione di ricorrenza
3) Qual è il problema risolto dall'algoritmo Algo?

Soluzione

1) $c(n) = 2 + c(n-2) + c(n-1)$
   L'algoritmo esegue due confronti (istruzioni 2 e 4) e due chiamate ricorsive, la prima su n-2 elementi (istruzione 3) e la seconda su n-1 elementi (istruzione 5).
   Pertanto la relazione di ricorrenza che conta il numero di confronti è:

$$c(n) = \begin{cases} 0, & n \leq 1 \\ c(n-1) + c(n-2) + 2, & n > 1 \end{cases}$$

2) La relazione c(n) ricorda quella dei numeri di Fibonacci che ha soluzione $O(2^n)$. Supponiamo quindi che $c(n) = O(2^n)$ e applichiamo il metodo di sostituzione per dimostrarlo.

$$\begin{aligned} c(n) &= c(n-1) + c(n-2) + 2 \\ &\leq c2^{n-1} + c2^{n-2} + 2 \\ &= c2^{n-2}(2+1) + 2 \\ &\leq c2^{n-2}4 + 2 \\ &= c2^n + 2 \end{aligned}$$

   Poiché non abbiamo ottenuto esattamente $c(n) \leq c2^n$, ma abbiamo un termine di ordine inferiore in più (2), facciamo un'assunzione più forte, ovvero assumiamo che $c(n) \leq c2^n - 1$ (che è $\leq c2^n$)
   In tal caso abbiamo:

$$\begin{aligned} c(n) &= c(n-1) + c(n-2) + 2 \\ &\leq c2^{n-1} - 1 + c2^{n-2} - 1 + 2 \\ &= c2^{n-2}(2+1) \\ &\leq c2^{n-2}4 \\ &= c2^n \end{aligned}$$

   Il passo base è verificato per $c \geq 1/2$.
3) L'algoritmo esegue l'ordinamento di n numeri (in tempo esponenziale).