Algoritmi e Strutture Dati con Laboratorio (modulo I)

Testo della prova scritta del 6 febbraio 2013

docente: Luciano Gualà

Cognome:..................... Nome:........................ Matr.:....... Corso di Laurea:................

Esercizio 1 [10 punti]

(a) Si ordinino le seguenti funzioni in ordine non decrescente di tasso di crescita asintotica. Per ogni coppia di funzioni $f_i(n), f_{i+1}(n)$ adiacenti nell'ordinamento si specifichi se $f_i(n) = \Theta(f_{i+1}(n))$ o se $f_i(n) = o(f_{i+1}(n))$.

Le funzioni sono: 2^{2n}, $\frac{n^2\sqrt{n^3+1}}{\sqrt{n+1}}$, $n^3 + \sqrt{n}\log^{10} n$, 2^n, $\frac{n^3+1}{\log n}$, $\frac{n^3+n^2}{\log\log n}$, $n^{\log n}$, $\frac{7n^{10}-\log n}{3}$, $\frac{n^3}{\sqrt{\log n}}$.

(b) Per un problema sono noti due algoritmi ricorsivi, A_1 e A_2 le cui complessità temporali sono descritte dalle seguenti equazioni di ricorrenza:

$T_1(n) = 2\,T_1(n-1) + 1$, $T_1(1) = 1$;

$T_2(n) = 16\,T_2(n/4) + \sqrt[3]{n \log n} + n^2$, $T_2(1) = 1$;

Dire, motivando la risposta, quale algoritmo è preferibile usare.

Esercizio 2 [12 punti] Sia V un vettore di n valori positivi. Progettare un algoritmo che, dato V, costruisca un *oracolo* (ovvero una struttura dati) che sia in grado di rispondere in tempo $O(1)$ a *query* (ovvero domande) del seguente tipo: dato un intero $k \in \{1, 2, \dots, n-1\}$, quale è la coppia di indici i e j, con $i \le k$ e $j > k$ che minimizza $V[i] + V[j]$?

L'algoritmo di costruzione dell'oracolo deve avere complessità temporale $O(n)$, mentre l'algoritmo di interrogazione dell'oracolo, come già detto, deve avere complessità temporale $O(1)$.

Esercizio 3 [13 punti] Si vuole dotare una superstrada di servizio wireless istallando opportuni access point. La superstrada è divisa in n tratte omogenee, numerate da 1 a n. Gli access point sono di due tipi, tipo *high* (H) e tipo *low* (L). Se un access point di tipo L è istallato nella tratta j, esso è in grado di coprire, oltre la tratta j, anche la tratta $j-1$; mentre un access point di tipo H copre le tratte $j, j-1, j-2$. I costi di tali istallazioni sono dei valori positivi c_j^L e c_j^H (che quindi dipendono dalla tratta e dal tipo dell'access point). Chiaramente, per ogni j, $c_j^L \le c_j^H$. Il problema che si vuole risolvere è quello di trovare un sottoinsieme di tratte su cui istallare opportuni access point in modo da coprire l'intera superstrada e contemporaneamente minimizzare il costo complessivo di istallazione. Progettare un algoritmo di programmazione dinamica con complessità temporale $O(n)$ che, dati i valori c_j^L e c_j^H, $j = 1, \dots, n$, calcola il valore di una soluzione di costo minimo per il problema.

Di seguito è riportato un esempio di istanza con $n = 6$ tratte. La soluzione ottima ha costo 18 e consiste nell'istallare due access point di tipo L nelle tratte 2 e 3, e uno di tipo H nella tratta 6.

	1	2	3	4	5	6
c^L	6	5	3	12	6	5
c^H	6	6	9	15	9	10