Algoritmi e Strutture Dati
a.a. 2009/10

Compito del 16/7/2010

Cognome: ______________________ Nome: ______________________

Matricola: ______________________ E-mail: ______________________

Parte I

(30 minuti; ogni esercizio vale 2 punti)

1. Determinare il numero di foglie e il numero totale di nodi in un albero binario completo in funzione dell'altezza *h* e dimostrare per induzione la correttezza della risposta.

2. Ordinare le seguenti funzioni in una lista in modo che se *f* viene prima di *g*, allora $f(n) = O(g(n))$. Se due o più funzioni hanno lo stesso ordine asintotico, lo si indichi esplicitamente.

n	2^n	$n \lg n$	n^3
n^2	$\lg n$	$n - n^3 + 7n^5$	$n^2 + \lg n$

3. Si enunci e si dimostri il teorema fondamentale della NP-completezza.

Algoritmi e Strutture Dati

a.a. 2009/10

Compito del 16/7/2010

Cognome: ______________________________ Nome: ______________________________

Matricola: ______________________________ E-mail: ______________________________

Parte II

(2.5 ore; ogni esercizio vale 6 punti)

1. Dato un vettore *v* di *n* interi non necessariamente distinti, si dice che un elemento è di **maggioranza** se appare in *v* almeno $\lceil n/2 \rceil$ volte. Scrivere due funzioni in C di complessità in spazio aggiuntivo costante e in tempo
   a. $\Theta(n^2)$
   b. $\Theta(n \log n)$

   per stabilire se *v* contiene un elemento di maggioranza, e in caso affermativo lo restituisca.

2. Dato un albero binario, progettare un algoritmo **ricorsivo** che costruisce un array bidimensionale *m* tale che, per ogni coppia di nodi *u* e *v*, l'elemento *m[u][v]* sia il minimo antenato comune di *u* e *v.* L'algoritmo deve richiedere tempo $O(n^2)$.
   Osservate che durante la ricorsione sul nodo corrente *u*, potete individuare quali coppie di nodi hanno *u* come minimo antenato comune.

   [Il **minimo antenato comune** di due nodi *u* e *v* è l'antenato comune di *u* e *v* che si trova più lontano dalla radice dell'albero.]

3. L'*arbitraggio* è un'operazione finanziaria per lucrare dalla differenza di prezzi tra le varie piazze e mercati. Sia $V = \{ v_1, v_2, \ldots , v_n \}$ un insieme di *n* valute e si indichi con C_{ij} il tasso di cambio tra le valute v_i e v_j (cioè, vendendo 1 unità di valuta v_i si ottengono C_{ij} unità di valuta v_j). Un arbitraggio è possibile se esiste una sequenza di azioni elementari di cambio (*transazioni*) che inizi con 1 unità di una certa valuta e termini con più di 1 unità della stessa valuta. Per esempio, se i tassi di cambio sono: 1.53 franchi svizzeri per 1 euro, 0.94 dollari americani per 1 franco svizzero, e 0.77 euro per 1 dollaro americano, possiamo convertire 1 euro in 1.1 euro, realizzando un guadagno del 10%. Si formuli il problema dell'arbitraggio come un problema (noto) di ricerca su grafi e si sviluppi un algoritmo efficiente per la sua risoluzione, discutendone correttezza e complessità.

4. Si stabilisca se le seguente affermazioni sono vere o false, fornendo una dimostrazione nel primo caso e un controesempio nel secondo:
   a. « Sia $G = (V, E, w)$ un grafo orientato e pesato, e sia $G' = (V, E, w')$ il grafo pesato ottenuto da *G* aggiungendo una costante *k* ai pesi (in altri termini, *G'* ha gli stessi vertici e gli stessi archi di *G*, e $w'(u,v) = k + w(u,v)$, per ogni arco (u,v) di $E'=E$). Allora, $p=<x_0,\ldots,x_q>$ è un cammino minimo in *G* se e solo se *p* è un cammino minimo in *G'*. »
   b. « Sia $G = (V, E, w)$ un grafo connesso non orientato e pesato, e sia $G' = (V, E, w')$ il grafo pesato ottenuto da *G* aggiungendo una costante *k* ai pesi (in altri termini, *G'* ha gli stessi vertici e gli stessi archi di *G*, e $w'(u,v) = k + w(u,v)$, per ogni arco (u,v) di $E'=E$). Allora, *T* è un albero di copertura minimo di *G* se e solo se *T* è un albero di copertura minimo di *G'*. »