

# Algoritmi e Strutture Dati con Laboratorio (modulo I)

Testo della prova scritta del 27 settembre 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:  $2^{0.22n}$ ,  $\frac{n\sqrt[4]{n^8+8}}{n+\sqrt{n+8}}$ ,  $n\sqrt[50]{n^{49}+50}$ ,  $2^n$ ,  $\frac{n^2}{\log n^{22}}$ ,  $\frac{n^2+212}{\log \log n}$ ,  $\frac{n^2+\sqrt{n}}{\log^{22} n}$ ,  $2^{22\sqrt{\log n}}$ ,  $n^2 + n^{1.5} \log^{1.5} n$ .

- (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:

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

$$T_2(n) = 2T_2(n/8) + \log^2 n + \sqrt[3]{n}, T_2(1) = 1;$$

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

**Esercizio 2 [12 punti]** Sia  $T$  un albero binario di  $n$  nodi con radice  $r$  in cui ogni nodo ha un valore non negativo associato. La *profondità* di un nodo  $v$  è il numero di archi del cammino da  $v$  alla radice. I nodi che si incontrano lungo tale cammino ( $v$  compreso) sono detti *antenati* di  $v$ . Diremo che un nodo  $v$  è *generazionalmente profondo* se la sua profondità è strettamente maggiore del valore di un suo antenato di valore minimo.

Si assuma che  $T$  è mantenuto attraverso una struttura collegata e che ogni nodo  $v$  abbia associato i seguenti campi: puntatori al padre e ai figli ( $v.p$ ,  $v.s$ ,  $v.d$ ) e valore del nodo ( $v.val$ ). Si progetti un algoritmo con complessità temporale  $O(n)$  che, preso  $T$ , restituisca il numero di nodi generazionalmente profondi di  $T$ . Si fornisca lo pseudocodice dettagliato dell'algoritmo.

**Esercizio 3 [13 punti]** Su una scacchiera  $n \times m$  è posizionato, nella casella in alto a sinistra, un robot. Il robot può muoversi sulla scacchiera spostandosi una casella alla volta, o una a destra o una in basso rispetto alla sua posizione corrente. Alcune celle della scacchiera contengono dei teschi, mentre altre delle monete. Se il robot finisce su una casella con il teschio scoppia, mentre se finisce in una casella con le monete è in grado di raccogliere queste monete. L'obiettivo del robot è quello di raggiungere la casella in basso a destra raccogliendo più monete possibile.

Siano  $M$  e  $T$  due matrici  $n \times m$ , dove  $M[i, j]$  contiene il numero di monete sulla casella  $(i, j)$  (riga  $i$ , colonna  $j$ ), e  $T[i, j] = 1$  se la casella  $(i, j)$  contiene un teschio, 0 altrimenti. Progettare un algoritmo di programmazione dinamica che calcoli il numero massimo di monete che il robot può raccogliere.