

Algoritmi e Strutture Dati (modulo I)
Testo della prova scritta del 24 febbraio 2020
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: $\frac{n^{4.01}+1}{\log^{10} n}$, 2^n , $n^4 + n^3 \log \log n$, $n^{3.99} \sqrt{\log n}$, $\frac{n^{4.01}}{\log^9 n}$, $\frac{n\sqrt{n^6+\log n}}{\sqrt{n^2+1}}$, $2^{n+\frac{1}{2}}$, $2^{\frac{n}{2}}$, $\frac{n^3}{\log \log n}$.

- (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:

$$T_1(n) = 2T_1(n/4) + \sqrt{n}, T_1(1) = 1;$$

$$T_2(n) = T_2(\sqrt{n}) + 1, T_2(1) = 1;$$

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

- Esercizio 2 [12 punti]** Sia A una matrice $n \times m$ di numeri. Si progetti un algoritmo con complessità lineare $O(mn)$ che costruisca una matrice $n \times m$ B dove per ogni $1 \leq i \leq n$ e $1 \leq j \leq m$, $B[i, j]$ contiene il minimo dei primi j elementi delle prime i righe, ovvero

$$B[i, j] = \min_{\substack{1 \leq h \leq i \\ 1 \leq k \leq j}} A[h, k].$$

Si fornisca lo pseudocodice dettagliato dell'algoritmo.

Esercizio 3 [12 punti]

- (a) A partire da un albero AVL vuoto, si mostrino le modifiche apportate all'albero in seguito ai seguenti inserimenti: 20, 30, 25, 22, 23, 24. Si cancelli poi il nodo con chiave 23.

- (b) Sia T un albero binario di n nodi in cui ogni nodo può essere di colore *bianco* o di colore *nero*. Si assuma che T è rappresentato tramite una struttura dati collegata in cui il record di un nodo v contiene le seguenti informazioni: il colore del nodo $c(v) \in \{B, N\}$ e due puntatori $s(v)$ e $d(v)$ rispettivamente al figlio sinistro e al figlio destro. Si progetti un algoritmo con complessità temporale $O(n)$ che calcoli il numero di foglie bianche di T . Si richiede inoltre che l'algoritmo non usi variabili globali. Si fornisca lo pseudocodice dettagliato.