

Cognome:..... Nome:..... Matr.:.....

Esercizio 1 [16 punti]

A: notazione asintotica. Dire quali delle seguenti relazioni asintotiche sono vere:

$$\begin{aligned} n + n \log^2 n &= o(n^{0.99} \log^{30} n); & \sqrt[3]{\log n} &= o(\log \log^{30} n); & n^2 &= \Omega\left(\frac{n^{2.001}}{\log^{2001} n}\right); & \frac{n\sqrt{n} + \log n}{\sqrt{n^3 + 3}} &= \Theta(\sqrt{n}); \\ 4^n &= \omega(3^n); & 2^n &= \Theta(2^n + n^3); & 2^{n/2} &= o(2^n + n^2); & 2^n &= \Theta(2^{n+2^4}); \end{aligned}$$

B: equazioni di ricorrenza. Fornire la soluzione asintotica alle seguenti relazioni di ricorrenza:

$$T(n) = 2T(n/2) + \sqrt{n}; \quad \text{Soluzione:}$$

$$T(n) = T(n-1) + n^3; \quad \text{Soluzione:}$$

C: algoritmi e complessità. Quale algoritmo useresti e quanto costa se devi:

- Cercare un elemento in una lista ordinata implementata con record e puntatori:
- Restituire gli elementi di un BST in ordine decrescente di chiavi:
- Capire in un grafo diretto e pesato quanto è lungo il cammino più corto da s a t che non usa una certa coppia di archi e_1 ed e_2 :
- Dato un grafo non diretto con nodi rossi e blu e due nodi s e t , dire se esiste un cammino da s a t che usa solo archi bicolorati (archi i cui estremi sono di colore diverso):

Esercizio 2 [8 punti]

Ti è dato in input un array $A[1 : n]$ di n numeri (si assuma per semplicità che n è pari). L'obiettivo è permutare i numeri in modo tale che non ci siano mai due numeri uguali adiacenti fra loro.

Progetta un algoritmo che risolve il problema (o dice correttamente che non è possibile) in spazio $O(n)$. Si argomenti con chiarezza la correttezza dell'algoritmo proposto e si fornisca l'analisi della complessità.

Esercizio 3 [8 punti]

Sia T un albero binario con n , dove ogni nodo v di T ha un colore $v.col \in \{1, 2\}$ associato. Diremo che un cammino radice-foglia è *alternato* se lungo il cammino non ci sono mai due nodi adiacenti dello stesso colore.

Si progetti un algoritmo che dato T restituisca il numero di cammini radice-foglia alternati che terminano su una foglia di colore 1 e il numero quelli che terminano su una foglia di colore 2. Si assuma che T è rappresentato tramite una struttura dati collegata, con record e puntatori, dove il record di ogni nodo, oltre al campo con il colore, contiene anche il puntatore al figlio sinistro e al figlio destro del nodo. L'algoritmo deve avere complessità $O(n)$. Si fornisca lo pseudocodice dettagliato.