

INFORMATICA GENERALE

Secondo Esonero e Compito Scritto

docenti: TIZIANA CALAMONERI, IVANO SALVO
Sapienza Università di Roma

10 giugno 2013

Coloro che intendono partecipare alla prova di esonero devono svolgere gli esercizi 2, 3 e 4; coloro che, invece, intendono partecipare all'intero compito scritto devono svolgere gli esercizi 1, 3 e 4.

Esercizio 1 (10 punti) Dato un vettore A di n numeri reali, progettare un algoritmo efficiente nel tempo e nello spazio che posizioni tutti gli elementi negativi prima di tutti gli elementi positivi.

Dell'algoritmo presentato:

1. **(3 punti)** Si scriva una descrizione a parole;
2. **(3 punti)** Si dia lo pseudocodice;
3. **(2 punti)** Si dettagli un esempio di funzionamento in cui n sia almeno 12;
4. **(2 punti)** Si valuti la complessità computazionale e si facciano le opportune osservazioni che consentano di dedurre che l'algoritmo è efficiente.

Esercizio 2 (10 punti) Si considerino le tre rappresentazioni in memoria di un grafo: matrice di adiacenza, liste di adiacenza e matrice di incidenza.

Dire qual è il tempo computazione per rispondere a ciascuna delle seguenti domande, giustificando le proprie affermazioni, su ciascuna rappresentazione:

1. **(2 punti)** due nodi sono adiacenti?

2. (2 punti) un nodo è estremo di un arco?
3. (2 punti) quanti sono gli archi nel grafo?
4. (4 punti) il grafo è connesso?

Esercizio 3 (10 punti) Scrivere una funzione C

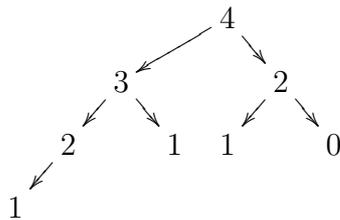
`lista camminoPiuLungo(tree T)`

che ricevuto in ingresso un albero T etichettato con valori interi, restituisca in una lista le etichette dei nodi contenute nel più lungo cammino radice-foglia in T .

Nel caso in cui ci siano più cammini della stessa lunghezza, tornare il cammino che termina nella foglia che compare prima in una qualsiasi visita in pre-ordine dell'albero.

Possibilmente scrivere una funzione lineare nel numero dei nodi di T , che effettua un'unica scansione di T , aiutandosi eventualmente con funzioni e parametri ausiliari.

ESEMPIO:



Ricevuto in input l'albero in figura, la funzione deve tornare la lista $\langle 4, 3, 2, 1 \rangle$.

Esercizio 4 (10 punti) Sia dato un heap massimo H con n nodi memorizzato in un vettore.

Si proponga a parole (2 punti) un algoritmo che trovi l'elemento minimo tra quelli memorizzati in H e se ne dia la complessità computazionale (1 punto).

Si trasformi poi l'algoritmo proposto sapendo che H viene memorizzato tramite record e puntatori, dandone lo pseudocodice (5 punti). Si valuti se e come cambia la complessità computazionale e si dia giustificazione della risposta (2 punti).