

Esame di Algoritmi e Strutture Dati

19 Marzo 2002

Si prega di scrivere il proprio nome, cognome e numero di matricola in alto a destra su ogni foglio. Si prega inoltre di svolgere le parti di teoria e laboratorio su fogli separati.

Teoria

1. Si consideri uno heap H in cui la chiave associata ad ogni nodo non è più piccola delle chiavi associate ai figli. Si proponga un algoritmo asintoticamente ottimale per costruire un nuovo heap H' con gli stessi elementi di H in cui la chiave associata ad ogni nodo non sia più grande delle chiavi associate ai figli. Si giustifichi la risposta.
2. Tra i dati relativi ad un'automobile vi sono il numero di telaio, la data di vendita, il colore, il prezzo di vendita, e l'età dell'acquirente. Si proponga una struttura dati per poter eseguire le seguenti operazioni in tempo al più logaritmico:
 - (a) Inserimento ed estrazione di un veicolo;
 - (b) Calcolo dell'età media di coloro che acquistano macchine di un colore c .
 - (c) Calcolo del prezzo medio delle auto blu vendute in un determinato periodo di tempo.
3. Un evento atmosferico ha provocato la distruzione di tutti i ponti di Venezia. La ditta Pinco Palla s.r.l. ci ha chiesto di calcolare la lista di ponti da riparare secondo le seguenti specifiche:
 - (a) Ogni punto di Venezia deve essere raggiungibile.
 - (b) Non devono esistere due percorsi alternativi per andare da un punto x ad un punto y .
 - (c) La riparazione di ogni ponte genera un punteggio che varia da ponte a ponte. La somma dei punteggi deve essere massima.

Si descriva formalmente il problema della ditta Pinco Palla e si proponga un algoritmo per risolvere il problema stesso.

Laboratorio

Si vuole realizzare un'implementazione della struttura dati hash table a indirizzamento aperto con doppio hashing per gestire un insieme di dati costituito da coppie (*CodiceFiscale*, *Reddito*), dove *CodiceFiscale* un oggetto di tipo **Stringa** e *Reddito* è un oggetto di tipo sconosciuto.

1. Si proponga una soluzione per il calcolo della posizione di inserimento di un elemento a partire dal *CodiceFiscale*. Si fornisca l'implementazione in Java della soluzione proposta.
2. Supponendo di voler mantenere un carico costante della tabella, si fornisca un algoritmo (e sua codifica in Java) per l'adeguamento della dimensione della tabella hash in caso di sovraccarico.