

008AA – ALGORITMICA E LABORATORIO

Esame del 9 settembre 2011

Cognome Nome:

N. Matricola:

Corso: A B

Esercizio 1. ($4+3+3$ punti) Si consideri la seguente variante del dizionario basato su tabella hash. Oltre alla tabella hash T a liste concatenate, gestita mediante una funzione hash H , viene utilizzato un vettore di appoggio V di p posizioni. La ricerca di una chiave k si svolge nel seguente modo: prima k viene cercata in V e, se appare nell' i -esima posizione di V , viene spostata in testa facendo così slittare in avanti tutti gli elementi delle prime $i - 1$ posizioni di V . Se invece k non viene trovata in V , allora si effettua la normale ricerca in T usando H : se la chiave appare in T , allora viene copiata in testa a V , facendo slittare in avanti tutti gli elementi di V (notare che, se V è pieno, l'ultimo elemento viene automaticamente scartato in questo modo); se non viene trovata neanche in T , allora la ricerca termina con fallimento.

1. Scrivere lo pseudocodice che realizza la suddetta operazione di ricerca.
2. Mostrare un esempio con chiavi numeriche, in cui la suddetta operazione effettua *meno* confronti della ricerca standard su tabella hash a liste concatenate (che utilizza solo T e H).
3. Mostrare un esempio con chiavi numeriche, in cui la suddetta operazione effettua *più* confronti della ricerca standard su tabella hash a liste concatenate (che utilizza solo T e H).

Esercizio 2. ($7+3$ punti) Si consideri un albero ternario, in cui ciascun nodo ha tre puntatori (sinistra **sx**, centro **cx** e destra **dx**) e un campo **chiave** numerico.

1. Scrivere un algoritmo ricorsivo che richiede tempo lineare nel numero di nodi per restituire il puntatore al nodo u tale che la somma delle chiavi discendenti dal figlio sinistro e di quelle discendenti dal figlio destro di u , sia minore della somma delle chiavi discendenti dal figlio centrale di u .
2. Motivare la correttezza e la complessità dell'algoritmo proposto.

Esercizio 3. ($7+3$ punti) Sia dato un grafo connesso, non orientato e non pesato $G = (V, E)$. Si definisca $N(u, k)$ come il numero di vertici in V che sono a distanza al più k da $u \in V$ (dove la distanza tra due vertici $x, y \in V$ è il minimo numero di archi in E che sono attraversati per andare da x a y).

1. Progettare un algoritmo che ricevuto in ingresso G e k , trova tutti i vertici $u \in V$ per cui $N(u, k)$ è massimo.
2. Motivare la correttezza e la complessità dell'algoritmo proposto.

Esercizio 3 bis [per studenti della classe 26]. ($3+7$ punti) Dimostrare che la complessità al caso pessimo del Quicksort è $O(n^2)$ tempo e che la sua complessità al caso medio è $O(n \log n)$ tempo (volendo, si può considerare la versione randomizzata del Quicksort).