

Algoritmi e strutture dati

Allievi della Laurea Specialistica in Ingegneria Informatica
Tema d'esame - 26 Giugno 2009 – ore 9.00-10.30

NOME: **COGNOME:**

MATRICOLA: **FIRMA:**

Il valore degli
esercizi è riportato
nel prospetto a lato.

Esercizio	1	2	3	4	5	6a	6b	6c	7a	7b
Valore	2	2	2	2	3	5	3	3	5	3
Valutazione										

Esercizi

- Definire i concetti di albero e foresta, puntualizzando la differenza fra gli stessi.
- Stabilire se la sequenza $\langle 23, 17, 14, 6, 13, 10, 1, 5, 7, 12 \rangle$ è uno heap binario, motivando la risposta.
- Illustrare l'esecuzione di RADIX-SORT sulla seguente lista di parole: DOG, RUG, DIG, BIG, NOW, BOX.
- Si consideri una tabella hash di lunghezza $m = 7$ gestita mediante hashing doppio con
$$h_1(k) = k \bmod m$$
$$h_2(k) = 1 + k \bmod (m - 1)$$
Simulare l'inserimento delle chiavi 10, 22, 31, 4, 15, 28, 17 nell'ordine dato.
- Un algoritmo ricorsivo impiega un tempo lineare per individuare il sottoproblema (del problema corrente) da affrontare mediante una chiamata ricorsiva (passo DIVIDE) e, una volta che tale chiamata è terminata, per utilizzare la soluzione da essa prodotta al fine di risolvere il problema corrente (passo COMBINA). La dimensione del sottoproblema affrontato dalla chiamata ricorsiva (passo IMPERA) è pari ai 9/10 della dimensione del problema corrente. Scrivere e risolvere l'equazione di ricorrenza relativa al tempo di esecuzione dell'algoritmo.
- Il problema del *conteggio delle lettere doppie di una stringa* è definito come segue.
Input: un vettore A contenente una sequenza di caratteri.
Output: un intero indicante il numero di occorrenze di due caratteri consecutivi identici in A (ad esempio, 4 per la stringa *bbaabbb*).
 - Utilizzare il paradigma divide et impera per scrivere lo pseudocodice di un algoritmo ricorsivo che risolva tale problema.
 - Analizzare la complessità dell'algoritmo proposto.
 - Ricordando che, a ogni nuova chiamata di una funzione ricorsiva, il contesto corrente della funzione (ovvero variabili locali e indirizzo dell'istruzione a cui ritornare) viene salvato su una pila, determinare la complessità spaziale dell'algoritmo proposto.
- Per profondità di un nodo di un albero si intende il numero di archi sul cammino che va dalla radice al nodo stesso. Si consideri il problema definito come segue.
Input: un albero ternario T (con radice $\text{root}[T]$) di n nodi ($n > 0$) e un intero k , $0 \leq k \leq n - 1$.
Output: il numero dei nodi di T che hanno profondità k .
 - Definire in pseudocodice una procedura ricorsiva che risolva tale problema.
 - Determinare la complessità dell'algoritmo proposto.