

Algoritmi e Strutture Dati

29 Giugno 2015

Cognome Nome Matricola

Domande

Domanda 1 (5 punti) Risolvere la ricorrenza $T(n) = 4T(n/2) + n \log n$ utilizzando il master theorem.

Domanda 2 (4 punti) Indicare il codice prefisso ottenuto utilizzando l'algoritmo di Huffman per l'alfabeto $\{a, b, c, d, e, f, g\}$, supponendo che ogni simbolo appaia con le seguenti frequenze.

a	b	c	d	e	f	g
16	12	2	8	3	9	6

Spiegare il processo di costruzione del codice.

Domanda 3 (5 punti) Scrivere una funzione `IsMaxHeap(A)` che dato in input un array di interi $A[1..n]$ che verifica se A è organizzato a max-heap e ritorna un corrispondente valore booleano. Valutarne la complessità.

Esercizi

Esercizio 1 (7 punti) Fornire lo pseudocodice di una procedura `min(A,B)` che dati due array A e B che siano una permutazione dell'altro ne trova l'elemento minimo confrontando esclusivamente elementi di A ed elementi di B (non si possono confrontare due elementi di A o due elementi di B tra loro, e non si possono fare copie degli array, ovvero la funzione deve operare con spazio costante). Valutare la complessità della funzione.

Esercizio 2 (11 punti) Dare un algoritmo per individuare, all'interno di una stringa $a_1 \dots a_n$ una sottostringa (di caratteri consecutivi) palindroma di lunghezza massima. Ad esempio, nella stringa “colonna” la sottostringa palindroma di lunghezza massima è “olo”. Più precisamente:

- dare una caratterizzazione ricorsiva della lunghezza massima $l_{i,j}$ di una sottostringa palindroma di $a_i \dots a_j$;
- tradurre tale definizione in un algoritmo (bottom up o top down con memoization) che determina la lunghezza massima;
- trasformare l'algoritmo in modo che permetta anche di individuare la stringa, non solo la sua lunghezza;
- valutare la complessità dell'algoritmo.