

INFORMATICA GENERALE

Esame Scritto

docenti: G. BONGIOVANNI, T. CALAMONERI, I. SALVO
Sapienza Università di Roma

8 luglio 2016

Esercizio 1 (10 punti) Date due matrici $n \times n$, l'algoritmo di Strassen, utilizzando la tecnica del *divide et impera*, ne calcola il loro prodotto con un costo computazionale caratterizzato dalla seguente equazione di ricorrenza:

$$\begin{aligned}T(n) &= 7T(n/2) + \Theta(n^2) \\T(1) &= \Theta(1).\end{aligned}$$

Si risolva tale equazione di ricorrenza utilizzando:

- (4 punti) il metodo iterativo;
- (3 punti) il metodo dell'albero;
- (1 punto) il metodo principale;

Rispetto all'algoritmo immediato che calcola il prodotto tra matrici utilizzando la sua definizione, l'algoritmo di Strassen lavora in un tempo asintoticamente inferiore, superiore o uguale? perché? si giustifichi brevemente la propria risposta (2 punti).

Esercizio 2 (10 punti)

- (6 punti) Supponete che due liste di interi in ordine non decrescente contengano la scomposizione in fattori primi di due numeri interi. Scrivere una funzione `C int mcdDaListaFattori(lista M, lista N)` che calcola il massimo comun divisore facendo il prodotto di tutti i numeri primi comuni presenti in M ed N .

ESEMPIO Se le liste di ingresso sono $\langle 2, 2, 2, 3 \rangle$ (scomposizione di 24) e $\langle 2, 2, 3, 3 \rangle$ (scomposizione di 36), la funzione deve tornare 12 (che effettivamente è l'MCD di 24 e 36), che la funzione dovrebbe calcolare come $2 \cdot 2 \cdot 3$, cioè il prodotto dei fattori primi comuni nelle due scomposizioni.

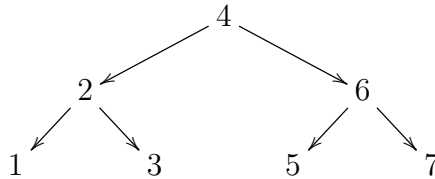
2. **(3 punti)** Scrivere una funzione `C lista fattoriP(int n, lista primi)` che genera la lista ordinata dei fattori primi di n usando nei test di divisibilità gli interi contenuti nella lista *primi*, che potete supporre essere una lista di interi ordinata crescente (abbastanza lunga!) con tutti i numeri primi necessari a scomporre n .

ESEMPIO Se n è 24, la funzione deve tornare la lista $\langle 2, 2, 2, 3 \rangle$. La lista *primi* sarà la lista $\langle 2, 3, 5, 7, 11, 13, 17, \dots \rangle$.

3. **(1 punto)** Usare le funzioni sopra descritte per scrivere una funzione `C int mcd(int m, int n, lista primi)` che restituisce il massimo comun divisore di m ed n .

Esercizio 3 (10 punti) Sia dato un vettore ordinato contenente esattamente $n = 2^h - 1$ elementi tutti distinti fra loro, per qualche intero $h > 1$. Progettare una funzione **ricorsiva** il più efficiente possibile che crea un albero binario di ricerca completo contenente tutte le chiavi del vettore.

Ad esempio, dato il vettore $A = [1, 2, 3, 4, 5, 6, 7]$, l'albero risultante deve essere:



Della funzione proposta:

1. **(3 punti)** Si dia la descrizione a parole;
2. **(3 punti)** Si scriva lo pseudocodice e si mostri la chiamata iniziale;
3. **(2 punti)** Si calcoli il costo computazionale possibilmente tramite la sua equazione di ricorrenza;
4. **(2 punti)** Si discuta brevemente sul modo in cui (eventualmente) cambiano l'algoritmo ed il costo computazionale nel caso in cui il vettore non sia ordinato.