ESAME DI ALGORITMI
Università degli Studi di Catania
Corso di Laurea Triennale in Informatica
28 marzo 2024

Si risolvano i seguenti esercizi in un tempo non superiore a 3 ore. Si abbia cura di consegnare la risoluzione dei primi 4 esercizi in un foglio (Foglio A) separato da quello utilizzato per la consegna degli ultimi due esercizi (Foglio B). Gli studenti delle vecchie coorti che devono sostenere solo il modulo di Algoritmi dovranno risolvere gli esercizi 1, 2, 3, 5 e 6 (tempo 2 ore). Gli studenti che devono sostenere solo il modulo di Laboratorio dovranno risolvere l'esercizio 4 (tempo un'ora).

——— Foglio A ———

1. Si supponga di operare su di un albero Rosso-Nero completo, contenente 15 chiavi. I nodi dell'albero sono tutti nodi neri ad esclusione dei nodi del livello 3 il cui colore è rosso. Nello specifico si effettuino 6 operazioni di cancellazione della radice dell'albero. Si fornisca la configurazione dell'albero dopo ciascuna delle 6 operazioni.

2. Sia $A = [5, 4, 2, 8, 10, 15, 9, 11, 12, 17, 14, 7, 16]$ un array di 13 elementi. Si supponga di eseguire l'ordinamento dell'array A mediante l'esecuzione dell'algoritmo HeapSort al fine di ottenere un ordinamento non decrescente dei suoi elementi. Si fornisca la configurazione dell'array dopo la fase di costruzione dello Heap e dopo ciascuna delle 12 iterazioni dell'algoritmo di ordinamento.

3. Limitandosi a quanto trattato a lezione, elencare gli algoritmi per il calcolo dei cammini minimi da sorgente singola, indicandone il nome, i limiti applicativi, e la relativa complessitá asintotica, sia nel caso in cui il grafo é implementato con matrici di adiacenza che nel caso in cui il grafo é implementato con liste di adiacenza.

4. Si forniscano gli pseudo-codici (o i codici in linguaggio `C/C++`) degli algoritmi di ordinamento Heap-Sort e Counting-Sort, compresi i codici delle eventuali procedure ausiliarie. Indicare anche la complessità computazionale delle procedure fornite, motivandone la risposta.

——— Foglio B ———

5. Si consideri l'equazione di ricorrenza $T(n) = aT\left(\frac{n}{3}\right) + n$. Si risolva l'equazione al variare del parametro reale $a \geq 1$, utilizzando il metodo Master. Si stabilisca per quali valori di b la soluzione $T(n)$ all'equazione soddisfa le seguenti condizioni:

   $$\text{(i.) } T(n) = \mathcal{O}(n) \qquad \text{(ii.) } T(n) = \Omega(n\log(n)) \qquad \text{(iii.) } T(n) = o\left(n^2\log(n)\right).$$

   Si disegni inoltre uno sketch dell'albero di ricorrenza associato all'equazione per $a = 3$.

6. Si consideri il seguente problema computazionale.

Matrix Chain Multiplication Problem
INPUT: $A_1, A_2, \ldots, A_n$, matrici tali che il numero di colonne di A_i sia uguale al numero di righe di A_{i+1}, per $1 \leq i < n$. **GOAL:** trovare una parentesizzazione di $A_1 \cdot A_2 \cdots A_n$ che minimizza il numero di moltiplicazioni scalari.

   Si dimostri che Matrix Chain Multiplication Problem ha la proprietá di sottostruttura ottima.