

Algoritmi e Strutture Dati II

15/09/08

Svolgere esattamente tre dei seguenti esercizi:

a) Sia dato il grafo $G=(V, E)$, dove $V=\{1, 2, 3, 4, 5, 6, 7\}$ ed $E=\{(1,2), (1,3), (4,2), (4,1), (5,5), (2,4), (3,5), (2,3), (5,6), (6,7)\}$. Rappresentare il grafo G utilizzando liste di adiacenza. Mostrare la lista di nodi ottenuta applicando l'algoritmo di visita in ampiezza a partire dal vertice 1.

b) Sia data la partizione $\{\{1, 2\} \{3\} \{4, 5\} \{6\} \{7\} \{8\} \{9, 10\}\}$. Mostrare come si modifica la foresta di alberi che la rappresenta in seguito all'applicazione delle funzioni:

| | | |
|--------------------------------------|-------------------------------------|-------------------------------------|
| <code>union(find(1), find(2))</code> | <code>union(find(1),find(3))</code> | <code>union(find(1),find(4))</code> |
| <code>union(find(1),find(5))</code> | <code>union(find(6),find(7))</code> | <code>union(find(7),find(8))</code> |
| <code>union(find(8),find(9))</code> | <code>union(find(9),find(3))</code> | |

utilizzando le euristiche note.

c) Sia dato il grafo non orientato e pesato $G=(V, E)$, dove $V=\{1, 2, 3, 4, 5, 6, 7\}$ ed $E=\{[1,2], [1,3], [4,2], [4,1], [3,5], [2,3], [5,6], [6,7]\}$, in cui gli archi $\{[1,2], [1,3], [4,2], [4,1]\}$ hanno peso 2, mentre gli altri hanno peso 4. Trovare il minimo albero ricoprente simulando l'esecuzione dell'algoritmo di Kruskal.

d) Siano date le matrici M_1, M_2, M_3, M_4 di dimensione rispettivamente $2 \times 5, 5 \times 8, 8 \times 5, 5 \times 2$. Trovare la disposizione ottimale delle parentesi che minimizza il costo del calcolo del prodotto $M_1 M_2 M_3 M_4$ ed il numero di moltiplicazioni elementari richieste utilizzando tale disposizione.

Rispondere ad esattamente due delle seguenti domande:

a) Illustrare e discutere l'algoritmo noto per costruire i cammini minimi tra tutte le coppie di vertici di un grafo orientato e pesato

b) Illustrare e discutere l'algoritmo noto per costruire la chiusura transitiva di un grafo orientato.

c) Enunciare e dimostrare il Teorema Fondamentale utilizzato per calcolare la più lunga sottosequenza comune (Sottostruttura ottima di una LCS).