Appello del 16 settembre 2004
Laboratorio di Algoritmi e Strutture Dati
Anno Accademico 2003/2003

Cognome e Nome: Docente:
Numero di Matricola:

Spazio riservato alla correzione

1	2	3	4	5	6	totale
/15	/15	/15	/15	/15	/15	/90

1. Scrivere lo pseudocodice dell'algoritmo di **Prim** per il calcolo del minimo albero ricoprente di un grafo (MST). Commentare l'uso delle strutture dati utilizzate. Indicare, giustificando la risposta, la complessità dell'algoritmo di **Prim**.

2. Provare il seguente teorema. Se si fa riferimento ad altri lemmi e/o corollari è necessario enunciarli.
   Teorema:
   Sia G=(V,E) un grafo connesso non direzionato con una funzione di peso *w* a valori reali definita su E. Sia A un sottoinsieme di E che è incluso in un minimo albero ricoprente per G, sia (S, V-S) un qualsiasi taglio di G che rispetta A, e sia (u,v) un arco leggero (*light edge*) che attraversa (S, V-S). Allora, l'arco (u,v) è un arco sicuro (*safe edge*) per A.

3. Aggiungere alla classe **LinkedTree** (implementa l'interfaccia **Tree** usando come nodo la classe **TreeNode**) il metodo: **CountNodes**() che restituisce il numero di nodi dell'albero che hanno almeno tre figli.

4. Scrivere una funzione **EstraiDaCoda** che ricevuto in input una coda Q di interi (**Integer**) ed un numero intero (**Integer**) n restituisca una coda R contenente tutti gli elementi di Q che sono multipli di n e maggiori di 27. Gli elementi di R devono comparire nello stesso ordine in cui comparivano in Q e alla fine della procedura, la coda Q deve contenere gli stessi elementi, nello stesso ordine, che conteneva in origine.

5. Aggiungere alla classe **ArrayStack** il metodo **multiPop**(int **num**) che restituisce in un array i primi num elementi presenti nello stack. Il metodo deve cancellare dallo stack gli elementi restituiti. Inoltre, deve lanciare l'eccezione **NotEnoughElements** se nello stack vi sono meno di **num** elementi.

6. Scrivere una funzione Java **PQ-Sort** che ricevuti in input una sequenza ed un comparatore restituisce in output una sequenza ordinata. La funzione **PQ-Sort**, per l'ordinamento, deve far uso di una coda a priorità. Discutere della complessità di tempo della funzione.