

**Laurea a Distanza in Ingegneria Informatica  
Algoritmi e Programmazione Avanzata**

## **Esame del 23 Luglio 2002**

### **Parte I (teoria)**

**Tempo: 45 minuti, NON è possibile consultare libri o appunti.**

#### **Esercizio 1 (3 punti)**

Si consideri una coda prioritaria implementata come heap. Si assuma che la priorità massima sia associata all'elemento con chiave minima. Sulla coda viene eseguita l'inserzione delle seguenti chiavi:

21 13 50 42 37 35 60 81 94 23 18

Si disegni la struttura dello heap risultante (sotto forma di albero).

#### **Esercizio 2 (3 punti)**

Si consideri un albero binario di ricerca inizialmente vuoto. Su di esso vengono eseguite in sequenza le operazioni di inserimento di elementi con le seguenti chiavi:

21 13 50 42 37 35 60 81 94 23 18 36

Si disegnino

- la struttura dell'albero risultante al termine degli inserimenti indicati
- la struttura dell'albero dopo la cancellazione dell'elemento con chiave 50.

#### **Esercizio 3 (3 punti)**

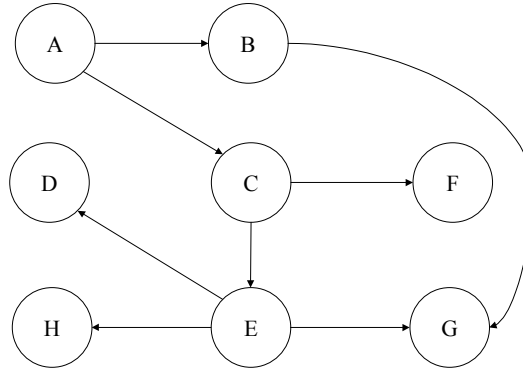
Si consideri una tabella di hash basata su open addressing che utilizza la tecnica del linear probing per risolvere le collisioni. La tabella è inizialmente vuota. Su di essa viene eseguita la seguente sequenza di operazioni di inserimento di elementi con chiave corrispondente ad un singolo carattere:

A M B Q R S Q U B C M C C D B E

Si mostri il contenuto della tabella di hash al termine della sequenza di inserimenti, assumendo che la sua dimensione sia pari a 19 e che la funzione di hash utilizzata corrisponda al numero d'ordine del carattere nell'alfabeto italiano (A=0, B=1, C=2, ecc.).

#### Esercizio 4

Sia dato il seguente grafo:



##### Punto a (3 punti)

Si disegnino la matrice di adiacenza e la lista di adiacenza del grafo.

##### Punto b (3 punti)

Si visiti il grafo in *profondità* ed in *ampiezza*, considerando A come vertice di partenza. Si indichi la sequenza di visita dei vertici nei due casi. Qualora necessario, si trattino i vertici secondo l'ordine alfabetico.

## Parte II (programmazione)

Tempo: 60 minuti, è possibile consultare libri o appunti.

Si consideri il grafo corrispondente alla rete dalle centrali di un sistema telefonico, ipotizzare che tutti i dati necessari siano già presenti in memoria.

Si scriva un programma C che esegua le seguenti operazioni.

- **Opzione A (max 12 punti).** Il programma verifica se esistono centrali *isolate*, ossia se esistono centrali che non possono essere raggiunte attraversando il grafo corrispondente alla rete telefonica. Nel caso esistano centrali isolate, il programma visualizza su schermo l'indicazione "Rete non connessa" ; in caso contrario il programma visualizza il messaggio "Rete connessa".
- **Opzione B (max 15 punti).** Il programma verifica se esistono centrali *isolate*, dopodiché se la rete è connessa, il programma verifica se esistono centrali *critiche*, ossia centrali che, se rimosse, creano centrali *isolate* (rendono la rete non connessa). Il programma visualizza su schermo l'indicazione "Rete resistente ad un guasto singolo" nel caso non esistano centrali *critiche*, oppure "Rete non resistente ad un guasto singolo".

Si motivi la struttura dati scelta.