

Algoritmi e Strutture Dati

Appello del 16 Settembre 2008

Modulo Teoria

Attenzione: Scrivere nome, cognome e numero di matricola sul foglio protocollo.

Esercizio 1 [Punti 10]

Dato un albero binario T i cui nodi contengono chiavi intere, progettare un algoritmo ricorsivo per stabilire quanti nodi di T sono 1-bilanciati e analizzarne la complessità.

Esercizio 2 [Punti 10]

La sequenza di chiavi $S = 101, 50, 20, 72, 3, 14, 25, 121, 115, 22, 73, 27, 40, 7$ deve essere inserita in una tabella hash di dimensione $m = 17$, inizialmente vuota. La tabella é a indirizzamento aperto e la scansione é eseguita per doppio hash:

$$h(k, i) = (k \bmod m + i \times 2^{k \bmod 5}) \bmod m.$$

Indicare per ogni chiave le posizioni scandite nella tabella e la posizione finale dove viene collocata.

Esercizio 3 [Punti 8]

Dato un grafo orientato memorizzato con le seguenti liste di adiacenza:

$$\begin{array}{lcl} 1 & \rightarrow & 2, 4, 7 \\ 2 & \rightarrow & 6 \\ 3 & \rightarrow & 1 \\ 4 & \rightarrow & 7 \\ 5 & \rightarrow & 2, 6 \\ 6 & \rightarrow & 3, 5 \\ 7 & \rightarrow & 5, 6 \end{array}$$

eseguire una visita BFS a partire dal vertice 1 indicando la sequenza dei vertici incontrati e l'albero BFS.

Esercizio 4 [Punti 2]

Discutere il caso ottimo di Quicksort e determinarne la complessità.

Modulo Laboratorio

Esercizio 1 [Punti 16]

1. Si commenti con javadoc il seguente codice, che implementa un algoritmo di visita post-ordine di un albero.

```
void visitaPostOrdine(Node nodo) {  
  
    if (nodo != null) {  
  
        Node figlio = nodo.figlio;  
  
        while (figlio != null) {  
  
            visitaPostOrdine(figlio);  
  
            figlio = figlio.fratello;  
  
        }  
  
        visita(nodo);  
    }  
}
```

ove si assume definita la seguente classe Node:

```
public class Node {  
    Object elemento;  
    Node padre,figlio,fratello;  
  
    public Node() {  
        elemento = null;  
        padre = figlio = fratello = null;  
    }  
    ...  
}
```

}

2. Definire il problema di trovare un albero di copertura minimo, e spiegare a parole l'algoritmo risolutivo che è stato implementato durante il corso.

Esercizio 2 [Punti14]

Si dia un'implementazione in codice java dell'algoritmo di ordinamento Quick-Sort.