Appello del 10 febbraio 2004
Laboratorio di Algoritmi e Strutture Dati – Prof. Carlo Blundo
Anno Accademico 2003/2003– Matricole Dispari-Pari

Cognome e Nome: Numero di Matricola:

Spazio riservato alla correzione

1	2	3	4	totale
/15	/15	/15	/45	/90

Non usare altri fogli, usare solo lo spazio sottostante. Fogli differenti da questo non saranno presi in considerazione per la correzione.

1. Aggiungere alla classe **LinkedTree** (implementa l'interfaccia **Tree** usando come nodo la classe **TreeNode**) il metodo **ObjectIterator nodes**(int k) che restituisce in output un iteratore sui nodi dell'albero che hanno grado al più k (hanno al più k figli). Una bozza della classe **TreeNode** è la seguente:

```
public class TreeNode implements Position {
  private Object element;  private TreeNode parent;  private NodeList children;
 …..
}
```

2. Sia **LogFile** l'implementazione dell'interfaccia **Dictionary** attraverso un'istanza LF della classe **NodeSequence**. Implementare il metodo public Object **findElement(**Object key**)** che restituisce l'elemento associato alla chiave key. Si supponga che le chiavi sono di tipo stringa e che due chiavi sono uguali se coincidono oppure se sono della stessa lunghezza. Implementare inoltre il metodo boolean **isEqualTo(**Object x, Object y**);** della classe **EqualityTester**. Discutere della complessità di tempo del metodo implementato.

3. Mostrare come implementare il tipo di dati astratto **Stack** usando solo una coda a priorità. (Scrivere la classe **StackPQ** che implementa l'interfaccia Stack usando come sola variabile d'istanza una variabile di tipo **PriorityQueue**).

4. Visita in ampiezza

4.1 Scrivere lo pseudocodice dell'algoritmo per la visita in ampiezza di un grafo (BFS). Commentare l'uso delle strutture dati utilizzate. [15 punti]

4.2. Indicare, giustificando la risposta, la complessità dell'algoritmo BFS. [15 punti]

4.3 Codificare l'algoritmo BFS in Java (scrivere una funzione Java che implementi l'algoritmo). Il codice deve far riferimento alle interfacce illustrate durante il corso. [15 punti]