

Cognome: Nome: Matricola:

UNIVERSITÀ DEGLI STUDI DELLA CALABRIA
Corso di Laurea in Ingegneria Informatica

Prova scritta di *Algoritmi e Strutture Dati* del 12 settembre '07
(durata della prova: 60 minuti)

Esercizio 1

Si consideri una classe *AlberoB* che rappresenta *alberi binari* in cui la parte informativa di ogni nodo è un numero intero. Si assuma che in tale classe siano implementati i seguenti metodi:

```
public interface AlberoB {  
    /* restituisce il sottoalbero destro dell'albero corrente, la complessità temporale è  $\theta(1)$  */  
    public AlberoB destro();  
  
    /* restituisce il sottoalbero sinistro dell'albero corrente, la complessità temporale è  $\theta(1)$  */  
    public AlberoB sinistro();  
  
    /* restituisce il valore memorizzato nella radice dell'albero, la complessità temporale è  $\theta(1)$  */  
    public int val();  
}
```

Si deve realizzare un metodo

```
public static boolean eRipetuto (AlberoB a, int x) {...}
```

che restituisce *true* se e solo se vi è almeno un nodo *n* nell'albero *a* tale che l'intero *x* appare sia nel sottoalbero sinistro che nel sottoalbero destro di *n*.

Si caratterizzi la complessità temporale e spaziale del metodo nel caso migliore e peggiore, specificando anche quali siano il caso migliore ed il caso peggiore per la complessità temporale e spaziale.

Caso Migliore:

Caso Peggior:

1. Complessità temporale: $\theta(\text{_____})$

1. Complessità temporale: $\theta(\text{_____})$

2. Complessità spaziale: $\theta(\text{_____})$

2. Complessità spaziale: $\theta(\text{_____})$

Commenti: _____

Esercizio 2

Dire quali delle seguenti affermazioni sono vere e quali false.

	<i>V</i>	<i>F</i>	<i>Affermazione</i>
1			La complessità intrinseca del problema di calcolare la somma di un array di interi è $\Omega(n)$, dove n è il numero di interi presenti nel array.
2			La complessità temporale del problema dell'inserimento in una array ordinato di interi è $\Omega(\lg n)$.
3			La funzione $f(n) = 2n \lg n^2$ è $O(n \lg n)$.
4			Sia G un grafo non orientato che contiene almeno un ciclo. Il numero di componenti connesse massimali è minore o uguale a $n-2$, dove n è il numero di nodi di G
5			Sia G un grafo pesato connesso, orientato, ed aciclico i cui pesi sono numeri positivi. l'algoritmo di Floyd potrebbe non essere in grado di calcolare le distanze minimi fra tutte le coppie di nodi di G
6			La complessità dell'inserimento di un elemento in una tabella hash in cui sono presenti n elementi nel caso peggiore è $\theta(n)$.
7			La complessità spaziale della visita infissa di un albero è $\Omega(n^2)$, dove n è il numero di nodi presenti nell'albero.
8			La complessità temporale dell'algoritmo di Dijkstra è $\theta(n^2)$, dove n è il numero di nodi presenti nel grafo, nel caso di rappresentazione con liste di adiacenza.
9			Per ogni coppia di nodi u,v appartenenti ad un grafo orientato debolmente connesso esiste sempre un cammino dal nodo u al nodo v e dal nodo v al nodo u .
10			Un grafo non orientato connesso e pesato (sugli archi) ammette sempre un unico albero ricoprenti di costo minimo.

Esercizio 3

Fornire la definizione formale di componente connessa massimale di un grafo non orientato.

[illegible]