

# Esame di Programmazione II

Appello di giorno 24 Giugno 2014  
Università degli Studi di Catania - Corso di Laurea in Informatica

## Testo della Prova

Si fornisca una classe C++ che implementi la seguente interfaccia `BST<T>`, che rappresenta un albero binario di ricerca e contenente i seguenti metodi virtuali. La corretta implementazione di ciascuno dei seguenti metodi permette l'acquisizione di 5 punti. La corretta implementazione della classe come template è facoltativa e permette l'acquisizione di ulteriori 3 punti:

1. `BST<T>*` `insert(T *x)` aggiunge un nuovo elemento all'albero e restituisce un puntatore ad un oggetto di tipo `BST<T>`;
2. `BST<T>*` `del(T *x)` cancella l'elemento `x` dall'albero, se presente, e restituisce un puntatore ad un oggetto di tipo `BST<T>`;
3. `int search(T *x)` restituisce 1 se `x` è presente nell'albero, 0 altrimenti (5 punti);
4. `void postorderPrint()` è una procedura che stampa in output gli elementi dell'albero secondo una visita postorder;
5. `void printLevel(int l)` stampa tutti i nodi dell'albero al livello `l`, ordinati in ordine crescente;
6. `void naturalFill(T *v)` è una procedura che, preso un array `v` con `n` elementi ordinati (dove `n` è anche il numero di nodi dell'albero) sostituisce le chiavi dei nodi dell'albero con gli elementi dell'array seguendo l'ordinamento naturale dettato dalla struttura dell'albero;

Si crei quindi un'istanza di `BST<int>` e si inseriscano al suo interno i seguenti valori:

```
23 4 6 8 12 21 5 9 7 3 16 2 24
```

Si chiami in seguito la procedura `naturalFill` passando come input l'array

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13].
```

Infine si invocano le procedure `postorderPrint()` e `printLevel(3)`.

L'output del programma sarà quindi:

```
1 2 4 6 8 10 11 9 7 5 3 13 12
1 4 7
```

```
template <class T> BST {
public:
    virtual BST<T>* insert(T *x) = 0;
    virtual BST<T>* del(T *x) = 0;
    virtual int search(T *x) = 0;
    virtual void naturalFill(T* v) = 0;
    virtual void postorderPrint() = 0;
    virtual void printLevel(int l) = 0;
}
```