

PROGRAMMAZIONE II – CANALE P-Z
ESONERO DEL 07/05/02

La prova dura 1h.

Si vuole definire una struttura dati astratta che implementa alberi binari i cui nodi contengono puntatori generici di tipo `void *`.

Le funzioni che si vogliono fornire sono:

1. una funzione che dato un puntatore p crea un albero formato da un solo nodo contenente il puntatore p ;
2. una funzione che, dato un puntatore p e due alberi binari t_0 e t_1 , crea l'albero binario che ha come radice un nuovo nodo contenente p e come figli gli alberi t_0 e t_1 ;
3. una funzione che dato un albero binario ritorna il valore del puntatore contenuto nella sua radice;
4. una funzione che, dato un albero t , ritorna 0 se si tratta di un nodo foglia ed un valore diverso da 0 altrimenti, inoltre, nel caso in cui non è una foglia, ritorna in due parametri i figli di t ;
5. una funzione che distrugge la radice di un albero, lasciando immutati gli alberi figli.

Esercizio 1. Dare una possibile implementazione del file `albbin.h` che costituisce l'interfaccia del modulo in cui sono implementate le suddette funzioni. (In pratica, si devono fornire i tipi necessari all'implementazione degli alberi binari e scrivere i prototipi delle funzioni sopra descritte.) **NB:** Inserire le linee di commento necessarie a chiarire lo scopo e l'uso delle funzioni.

Esercizio 2. Supponendo di conoscere solo le funzioni definite dall'interfaccia `albbin.h` e di non avere dettagli sull'implementazione effettiva degli alberi binari (ovvero di non sapere se questi sono ottenuti con record e puntatori o meno, e nel caso siano implementati con record e puntatori di non conoscere i campi della struttura e/o di non sapere come sono usati), scrivere una funzione che dato un albero binario t crea un nuovo albero ottenuto scambiando tra loro la posizione dei figli dell'albero.

Esercizio 3. Sempre usando solo le funzioni in `albbin.h`, scrivere una funzione che distrugge interamente un albero binario.