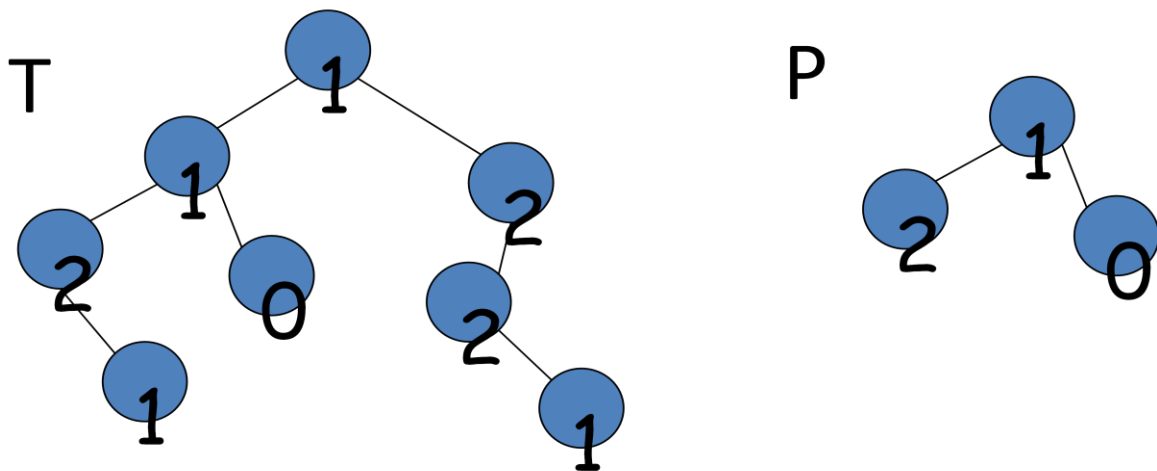


## Esercizio 4 del 4/3/2014 consegna libera

Si tratta di eseguire un pattern matching in cui sia il testo che il pattern sono alberi binari. La richiesta è di scrivere una funzione che dati un albero T ed un altro albero P risponda true sse c'è un match di P in T. Cosa significa un match tra 2 alberi è spiegato nel seguente esempio

**Esempio:** Usiamo come T lo stesso esempio dell'esercizio 2 e 3, mentre P è l'albero a destra.



C'è un match di T in P a partire dal nodo caratterizzato dal cammino 0. Infatti questo nodo in T è radice di un sottoalbero in cui la parte superiore (quella attaccata alla radice) è identica a P. In questo caso la funzione che realizza il match dovrebbe restituire il valore true seguito da 0 per indicare il cammino dalla radice di T alla radice del match. Sottolineiamo che i match tra alberi sono sempre contigui.

Per risolvere il problema appena descritto, si chiede di realizzare una funzione:

`bool cerca_match(nodo*T, nodo*P, int*C, int &lung)` che soddisfa la seguente pre- e post-condizione:

PRE\_cerca=(T e P alberi corretti, C array con almeno tanti elementi quanta è l'altezza dell'albero)

POST\_cerca=(restituisce true sse in T esiste un match di P e, in caso ci sia un match, C[0..lung-1] è il cammino per arrivare dalla radice di T alla radice del match minimo di P in T).

In POST\_cerca quando si parla di match minimo, si fa riferimento al fatto che ogni match è caratterizzato da un cammino che dalla radice di T arriva alla radice del match. Quindi l'ordine tra i match è determinato da quello sui corrispondenti cammini e questo ordine è quello definito all'inizio dell'esercizio 3 con una piccola aggiunta dovuta al fatto che ora si devono considerare anche cammini che non arrivano necessariamente ad una foglia. Questo fatto comporta che ora si deve considerare anche il caso di confrontare un cammino  $p1=i_1, \dots, i_k$  ed un altro cammino  $p2=p_1, i_{(k+1)}, \dots, i_{(k+m)}$ , cioè  $p2$  è  $p1$  con qualche nodo in più. In questo caso  $p1$  è minore di  $p2$ .

**Output da produrre:** su "output" va stampato il booleano prodotto da `cerca_match` e, nel caso esso sia true (1), si deve anche stampare il cammino che conduce al match trovato che la funzione avrà costruito in C.

**Correttezza:** Dimostrate la correttezza di `cerca_match`.