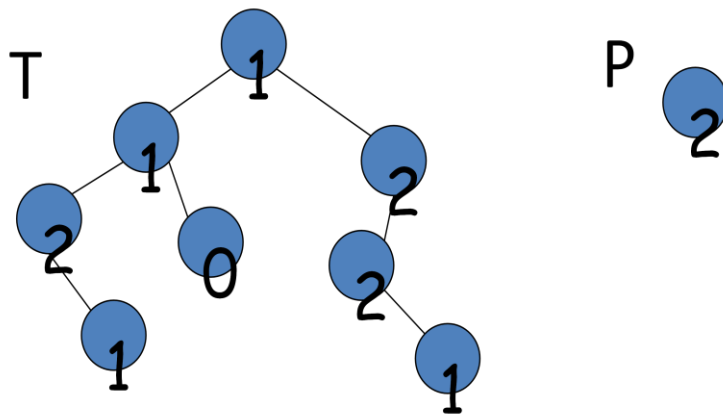


Esercizio 5 del 5/3/2014 consegna libera

E' un estensione dell'esercizio 4 in cui anziché calcolare il primo match di P in T, vogliamo calcolare tutti i match. Quello che vogliamo è calcolare in un array CT tutti i cammini che conducono in T alle radici di tutti i match di P in T elencati in modo crescente rispetto all'ordine tra cammini definito nell'esercizio 4. Spieghiamo meglio cosa si deve fare usando di nuovo l'albero T dell'Esercizio 4 mentre il pattern consiste di un unico nodo con 2.

Esempio. Ci sono 3 match di P in T ed i cammini corrispondenti sono 00, 1, e 10. Nell'array CT si deve scrivere questi 3 cammini e per distinguerli tra loro è necessario separarli con qualche simbolo che non può appartenere ai cammini. A questo scopo usiamo -1 e quindi in questo esempio CT dovrebbe contenere [0,0,-1, 1, -1, 1,0, -1].



La funzione che esegue questo calcolo è la seguente:

bool cerca_ogni_match(nodo*T, nodo*P, int*C, int*CT, int & lung, int & lungct) che soddisfa le seguenti pre- e post condizione:

PRE=(T e P alberi corretti, C e CT hanno n. di elementi sufficienti, lung e lungct sono ≥ 0 e C[0..lung-1] e CT[0..lungct-1] sono definiti)

POST=(la funzione restituisce true sse c'è almeno un match di P in T e in questo caso CT[0..lungct-1] contiene tutti i cammini dei match di P in T elencati in ordine crescente e separati da -1)

L'affermazione nella PRE che C e CT hanno abbastanza elementi significa che non ci si deve preoccupare di uscire dai limiti di questi array.

Il main è tale da stampare su "output" il booleano restituito ed eventualmente il contenuto di CT.

Correttezza: dimostrare la correttezza di cerca_ogni_match.