

Compitino 2 di Programmazione del 9/6/2015

Dato un albero binario non vuoto, un punto d'innesto è un nodo dell'albero che abbia almeno uno dei due puntatori left e right uguale a 0. Come il nome suggerisce, se il nodo n è un punto d'innesto allora è possibile aggiungere un nuovo nodo all'albero come un figlio di n (un figlio non già presente).

Esempio 1: si consideri l'albero $R = 2(3(_, 4(_, _)), 2(3(_, _), _))$ certamente le foglie con valore 4 e 3 caratterizzate dai cammini, $(0,1)$ e $(1,0)$, sono punti di innesto che consentono di aggiungere 2 nodi ciascuno (dato che in essi mancano entrambi i figli), ma anche i 2 figli della radice (con valori 3 e 2) hanno solo 1 figlio e quindi anch'essi sono punti di innesto.

1) La prima parte dell'esercizio richiede di calcolare una lista che descriva tutti i punti di innesto di un albero in ordine infisso. Come nodo di questa lista si deve usare la seguente struttura:

```
struct innesto{ bool l,r; nodo*N; innesto* next;}
```

Esempio 2: la lista dei punti d'innesto dell'albero dell'Esempio 1 è la seguente:

$(l=true, r=false, N=(0)) \rightarrow (l=true, r=true, N=(0,1)) \rightarrow (l=false, r=true, N=(1)) \rightarrow (l=true, r=true, N=(1,0))$

si osservi che i campi N dovrebbero avere come valore dei puntatori a nodi di R , ma per poter comunque descrivere la lista dei punti di innesto, nei campi N abbiamo inserito i cammini che dalla radice di R conducono ai nodi puntati dai campi N .

Si chiede quindi di realizzare una funzione $f0$ che rispetta la seguente specifica:

PRE=(albero(R) corretto)

innesto* $f0(\text{nodo}*R)$

POST=(restituisce la lista dei punti d'innesto di albero(R) in ordine infisso)

2) Si chiede di realizzare una funzione ricorsiva $f1$ che rispetta la seguente specifica:

PRE=(lista(Inn) è corretta, in particolare ciascun nodo punta ad un corrispondente punto d'innesto di un qualche albero, INP contiene m interi seguiti dalla sentinella -2 , $0 \leq m$, sia $vInn$ il valore iniziale di lista(Inn))

int $f1(\text{innesto}* \& Inn, \text{ifstream } \& INP)$

POST=(se $vInn$ contiene n campi l/r a true, allora vengono letti $x = \min(m, n)$ valori da INP e aggiunti x nuovi nodi usando i punti di innesto di $vInn$ in ordine; in caso resti una parte di $vInn$ non usata (succede quando $n > m$), questa lista rimasta deve essere il valore finale del parametro Inn . La parte di $vInn$ che è usata viene deallocata)

Attenzione: i punti di innesto che corrispondono alle foglie, consentono di aggiungere 2 nuovi nodi. E' possibile che la sentinella -2 venga trovata su INP dopo aver aggiunto il figlio sinistro della foglia., ma prima di aggiungere quello destro. In questo caso il punto di innesto va restituito con $l=false$ e $r=true$.

Correttezza : dimostrare la correttezza di $f0$ rispetto a pre e post date.

