

Esame di programmazione del 17/9/2019 Parte ricorsiva

Esempio 1: L'esercizio concerne il pattern matching su un albero BST. Dato un BST, per esempio il seguente:

$R = [20]([10]([5]([0](_, _), [12](_, _)), [15]([13](_, _), _)), [30]([25]([22](_, _), _), [35]([33](_, _), _)))$

e un pattern $P = [25, 22, 35, 38]$. Solo i primi 3 elementi del pattern trovano match in R e sono tutti nel sottoalbero destro della radice. I 3 nodi sono sottolineati nella seguente rappresentazione dell'albero:

$R = [20]([10]([5]([0](_, _), [12](_, _)), [15]([13](_, _), _)), [30]([25]([22](_, _), [35]([33](_, _), _)))$

Il match di P va cercato su tutti i nodi dell'albero R e i nodi vanno considerati nell'ordine prefisso (radice, sinistra, destra). Si osservi che, in generale, non si deve percorrere tutto l'albero, infatti la proprietà BST può essere usata per evitare di visitare sottoalberi in cui sicuramente non ci saranno match del prossimo elemento del pattern. Questo fenomeno è mostrato nell'esempio precedente in cui la proprietà BST può permettere di evitare di cercare $P[0]=25$ nel sottoalbero sinistro della radice R.

Esercizio da fare: si chiede di realizzare una funzione **ricorsiva** match che per R e P come in Esempio 1 restituisca col return una lista di 3 nodoP che puntano ai 3 nodi sottolineati dell'Esempio 3 e restituisca con i parametri P e dimP passati per riferimento dimP=1 e P=[38] cioè restituisca quanto del pattern iniziale resti ancora da matchare.

La funzione match deve soddisfare la seguente specifica:

PRE=(albero(R) è benformato, profondità ≥ 0 , dimP ≥ 0 e P ha dimP elementi)

nodoP* match(nodo*R, int profondità, int*&P, int & dimP)

POST=(restituisce una lista di nodoP che corrisponde al massimo match di P nei nodi di R esaminati in ordine prefisso, la lista, P e dimP sono come descritti prima)

Avviso: match dovrebbe usare la proprietà BST per evitare operazioni inutili. Nel programma dato c'è anche la funzione conc che concatena 2 liste di nodoP anche vuote.

Esempio 2: Si consideri di nuovo R dell'Esempio 1 e prendiamo ora $P = [30, 15, 13]$. Il primo match di $P[0]=30$, è nel sottoalbero destro della radice di R e, visto che l'albero va percorso in ordine prefisso, non sarà più possibile ritornare al sottoalbero sinistro per trovare 15 e 13 che ci sarebbero. Quindi match dovrà restituire una lista con 1 solo nodoP che punta al nodo 30 e ha campo prof=1, dimP=2 e $P = [15, 13]$.

Correttezza: Dimostrare induttivamente la correttezza di match rispetto alle PRE e POST date.