

## Secondo compito di Programmazione dell'11/06/2021

L'esercizio riguarda il match di un pattern sulla frontiera di un albero binario T. La frontiera di un albero è la sequenza delle sue foglie prese da sinistra a destra.

**Esempio:** supponiamo di avere l'albero T seguente:

1(2(4(8(, ,), ), 6(0(, ,), ), 3(5(9(, ,), ), 7(, ,)))

la sua frontiera consiste di 4 foglie con i seguenti campi info: 8, 0, 9, 7. Se il pattern da matchare è  $P=[0,7,9]$ , sulla frontiera di T troviamo il match di  $P[0]$  e  $P[1]$  solamente. Quindi cerchiamo un match di P nella frontiera di T che non deve essere necessariamente né contiguo ( $P[0]$  viene trovato nella seconda foglia, mentre  $P[1]$  viene trovato nella quarta foglia) né il match deve essere completo (non matchiamo tutto P). Si matcha  $P[0]$  con la prima foglia (da sinistra) con  $\text{info}=P[0]$ , poi si cerca di matchare  $P[1]$  partendo dalla foglia immediatamente successiva e continuando a tentare finché non si trova una foglia con  $\text{info}=P[1]$ , e così via.

Ovviamente può anche succedere di trovare un match contiguo e completo, ma si devono considerare anche match incompleti e non contigui.

Vediamo un altro albero T:

1(2(4(8(, 11(, ,)), 15(, ,)), 6(0(, ,), 7(, ,))), 3(5(17(, ,), 9(, ,)), 70(, ,)))

Questo albero ha 7 foglie con i seguenti campi info: 11, 15, 0, 7, 17, 9, 70. Per  $P=[0,7,9]$ , come prima, abbiamo un match non contiguo, ma completo di P sulla terza, quarta e sesta foglia della frontiera. Ripetiamo che, in generale, non è richiesto che si trovi un match completo.

Si chiede di scrivere una funzione ricorsiva matchF con il seguente prototipo:

```
void matchF(nodo*T, int* P, int dimP, int & im, nodo* L[])
```

e che soddisfi le seguenti specifiche :

PRE=(alb(T) è ben formato  $0 \leq \text{im} \leq \text{dimP}$ , P ed L hanno dimP posizioni,  $\text{vim}=\text{im}$ )

POST=( $L[\text{vim}..\text{im}-1]$  contiene puntatori alle foglie di T che matchano  $P[\text{vim}]..P[\text{im}-1]$  in modo che elementi successivi di L puntano a foglie successive di T (da sinistra a destra))

**Attenzione:** notare che il parametro im è passato per riferimento. Notare anche che L è un array che contiene valori di tipo nodo\*.

**Correttezza:** scrivere la dimostrazione di correttezza di matchF rispetto alle PRE e POST date

**Avvertenza:** si chiede di scrivere SOLO la funzione ricorsiva matchF (non serve il main, funzioni di stampa o cose del genere). D'altra parte matchF può usare delle funzioni ausiliarie per raggiungere il suo scopo. In questo caso, queste funzioni devono essere ricorsive anch'esse e devono avere PRE e POST.