

Esercizio 0 del 29/5/2017

Questo esercizio assomiglia molto all'esercizio 3 della settimana del 22/5 e riguarda il pattern matching di un pattern P ($\text{char } P[0..\text{dim}P-1]$) in un testo T che è una lista concatenata composta da nodi del solito tipo:

```
struct nodo{char info; nodo* next;};
```

A differenza dell'esercizio 3 del 22/5, ora il match non deve essere necessariamente contiguo, cioè è possibile fare match di $P[i]$ su un certo nodo n di T , e poi trovare il match di $P[i+1]$ in un qualunque nodo di T che sia successore di n . Come nell'esercizio 3 del 22/5, si cerca il match di tutto P . Quindi, solo in caso si trovi un match completo di P in $L(T)$, si deve staccare da $L(T)$ i nodi sui quali si è trovato il match di P e si deve restituire la lista di questi nodi (chiamiamo questa lista la lista del match) e si deve anche restituire la lista dei nodi di $L(T)$ che non hanno contribuito al match (chiamiamo questa lista quella restante). Vediamo un esempio.

Esempio 1. Sia $L(T) = a \rightarrow l \rightarrow b \rightarrow e \rightarrow r \rightarrow o \rightarrow b \rightarrow e \rightarrow l \rightarrow l \rightarrow o$ e $P = ['b', 'l', 'o', 'l']$. In questo caso c'è un match, con 'b' che matcha il terzo nodo di $L(T)$, 'l' il nono nodo e 'o' l'ultimo nodo. Quindi la lista del match sarebbe, $b \rightarrow l \rightarrow o$ e la lista restante sarebbe, $a \rightarrow l \rightarrow e \rightarrow r \rightarrow o \rightarrow b \rightarrow e \rightarrow l$. In caso $P = ['b', 'l', 'o', 'l']$ il match fallirebbe e quindi la lista restante sarebbe semplicemente $L(T)$ senza cambiamenti, mentre la lista del match sarebbe 0.

La funzione ricorsiva che deve fare queste operazioni è:

```
nodo* match (nodo* & T, char * P, int dimP)
```

che deve rispondere alle seguenti pre e postcondizioni:

PRE=($L(T)$ è corretta, $\text{dim}P > 0$, P ha $\text{dim}P$ elementi definiti)

POST=(se c'è un match di P in $L(T)$ allora la funzione restituisce col return la lista del match e in T la lista restante, altrimenti, restituisce 0 col return e T punta alla stessa lista cui puntava all'inizio).

Viene dato un main che si occupa delle letture e della costruzione di T e anche delle stampe finali.

Correttezza: dimostrare induttivamente la correttezza di match rispetto a PRE e POST date.