

### Esercizio 1 del 10/5/2021

Si tratta di un esercizio di pattern matching nel quale sia il testo T che il pattern P sono liste concatenate. Diremo che in P c'è un match di P quando  $T=T1@P'@Resto$ . con  $P'$  tale che ha la stessa lunghezza di P e i campi info dei nodi di  $P'$  sono uguali a quelli dei corrispondenti nodi di P. Nel caso ci sia un match di P in T, vogliamo restituire  $P'$  e anche la parte di T che resta dopo che  $P'$  è stata estratta da T.

Vogliamo fare questa operazione in 2 modi, seguendo le tipologie 1 e 3 delle funzioni ricorsive che abbiamo già discusso la settimana scorsa.

Per quanto riguarda la tipologia 1, useremo la struct:

struct `doppial`{`nodo*L,*S; doppial(nodo*a=0,nodo*b=0){L=a; S=b;}}`; e si richiede la funzione `match1` che soddisfa le seguenti specifiche:

PRE=(T e P sono liste ben definite)

`doppial match1(nodo*T, nodo*P)`

POST=(se in T c'è un match  $P'$  di P, la funzione restituisce un valore `doppial q`, con  $q.S=P'$  e  $q.L$ =quello che resta di T una volta tolto  $P'$ , si considera il match  $P'$  che inizia più a sinistra in T)&&( se in T non c'è match di P, allora la funzione restituisce un valore `doppial q` con  $q.L=T$  e  $q.s=0$ )

La seconda funzione segue la tipologia 3 e restituisce  $P'$  con il return e quello che resta di T usando il parametro passato per riferimento. `match3` deve soddisfare le seguenti specifiche:

PRE=(T e P sono ben formate)

`nodo* match3(nodo*&T, nodo*P)`

POST=(se c'è match restituisce il primo match  $P'$  col return e quello che resta di T nel parametro T) &&(se non c'è match T resta invariata e ritorna 0 col return).

**Correttezza:** dimostrare la correttezza delle 2 funzioni rispetto alle PRE e POST date.