

## Esercizio 1

L'esercizio è semplice: data una lista concatenata L e un valore x, vogliamo costruire una nuova lista L1 che contiene gli stessi nodi di L a parte il primo nodo di L con info=x. Se nessun nodo di L ha info=x, allora L=L1. Sottolineo il fatto che L1 debba essere costruita con gli stessi nodi che componevano L.

**Esempio:** se L = 2 -> 1 -> 1 -> 3 -> 1 -> 0 -> 2 e x=1, allora L1 = 2 -> 1 -> 3 -> 1 -> 0 -> 2

Si chiede di realizzare questa operazione seguendo i 3 diversi tipi di soluzioni ricorsive visti nelle lezioni. Precisamente si chiede di realizzare le seguenti 3 funzioni:

- 1) Soluzione di tipo 1  
nodo\* elimOne(nodo\*L, int x)  
con  
PRE=(Lista(L) ben formata)  
POST=(restituisce una lista che contiene tutti i nodi di Lista(L) a parte il primo con info=x, che, se esiste, va anche deallocato).
- 2) Soluzione di tipo 2  
void elimOne2(nodo\*L, int x)  
con  
PRE=(Lista(L) ben formata e non vuota e col primo nodo tale che il suo campo info non sia x,  
vLista(L)=Lista(L))  
POST=(Lista(L) contiene tutti i nodi di vLista(L) a parte il primo con info=x, che, se esiste, va anche deallocato).
- 3) Soluzione di tipo 3  
void elimOne3(nodo\*&L, int x)  
con  
PRE=(Lista(L) ben formata e vLista(L)=Lista(L))  
POST=(Lista(L) è come vLista(L) a parte il primo nodo con info=x, che, se esiste, va anche deallocato)

Si osservi il fatto che elimOne2 ha una PRE più restrittiva delle altre 2. Questa restrizione assicura che il puntatore L non cambia di valore durante l'esecuzione della funzione. Cioè che il primo nodo di Lista(L) e vLista(L) è lo stesso. Questo fatto permette a elimOne 2 di restituire void. Invece per elimOne3, anche se L cambiasse, il passaggio per riferimento penserebbe a comunicare la cosa al chiamante.

I test automatici dell'esercizio prevedono che ogni volta si usino le 3 funzioni. Il main costruisce 3 liste, L, L2 e L3 dove L e L3 sono uguali, mentre L2 soddisfa la PRE di elimOne2. Quindi L e L3 devono essere usate per elimOne ed elimOne3, mentre L2 serve per elimOne2.

### Correttezza:

Dimostrare la correttezza di elimOne2 con una prova induttiva.