

### Esercizio 1 del 7/6/2021

Sono esercizi sulle liste concatenate. I nodi delle liste hanno un campo info intero. Data una lista L, un  $k \geq 0$  e un intero y, si vogliono costruire 2 funzioni, elimI e elimF che verificano che L contenga almeno k nodi con info=y e in questo caso eliminano rispettivamente i primi k nodi con info=y di L (I sta per inizio) e gli ultimi k nodi di L con info=y (F sta per fine). Vediamo un esempio,

**Esempio.** Sia la lista L = 0 -> 1 -> 3 -> 2 -> 0 -> 1 -> 0 -> 2 -> 3 -> 0 -> 1 -> 0 -> 2 -> 1 -> 0. Sia k=4 e y=0.

L contiene 6 nodi con info=0. Quindi elimI deve modificare la lista L originale nella seguente lista:

1 -> 3 -> 2 -> 1 -> 2 -> 3 -> 1 -> 0 -> 2 -> 1 -> 0, mentre elimF deve produrre la seguente lista:

0 -> 1 -> 3 -> 2 -> 0 -> 1 -> 2 -> 3 -> 1 -> 2 -> 1. Insomma elimI elimina i primi 4 nodi con info=0 di L (e lascia gli ultimi 2 nodi), mentre elimF elimina gli ultimi 4 nodi con info=0 (e lascia i primi 2). Con la stessa lista L, se k fosse 3 e y=3, non contenendo L 3 nodi con y=3 (ma solo 2), le 2 funzioni non cambierebbero L.

Le 2 funzioni devono avere il seguente prototipo e soddisfare le seguenti PRE e POST:

PRE=(L(L) ben formata,  $k \geq y$ ), vL=L)

nodo\* elimI(nodo\*L, int &n, int k, int y)

POST=(se vL contiene almeno k nodi con info=y, allora L è ottenuta da vL eliminando i primi k nodi con info=y di vL, altrimenti, L=vL) &&( se nodi di vL vengono cancellati, essi vanno deallocati)

La funzione elimF ha specifiche uguali con la sola differenza che nella POST si richiede l'eliminazione degli ultimi k nodi con info=y.

**Attenzione:** il parametro n, durante l'andata della ricorsione, serve a contare i nodi con info=y. Il passaggio per riferimento, serve a fare in modo che n possa servire a decidere cosa vada fatto durante il ritorno della ricorsione.