

### Esercizio del 16-5-2018

Data una lista concatenata  $Q$ , un intero  $k > 0$  e un secondo intero  $y$ , vogliamo una funzione ricorsiva che faccia le seguenti operazioni:

- a) se  $L(Q)$  contiene almeno  $k$  nodi con campo  $\text{info}=y$ , elimina gli ultimi  $k$  di questi nodi,
- b) se invece  $L(Q)$  contiene meno di  $k$  nodi con  $\text{info}=y$ , allora  $L(Q)$  non cambia.

**Esempio 1:**  $L(Q) = -1 \rightarrow 2 \rightarrow -1 \rightarrow 4 \rightarrow -1 \rightarrow 0$ ,  $k=2$  e  $y = -1$ .  $L(Q)$  contiene 3 nodi con  $\text{info}=-1$  e quindi siamo nel caso (a), per cui gli ultimi 2 nodi con  $\text{info} = -1$  vanno eliminati, e  $L(Q)$  alla fine deve diventare,  $-1 \rightarrow 2 \rightarrow 4 \rightarrow 0$ . I due nodi staccati dalla lista vanno deallocati. Se  $k=4$  e  $y=-1$ ,  $L(Q)$  non dovrebbe cambiare.

La funzione ricorsiva richiesta deve soddisfare le seguenti specifiche:

PRE= (  $L(Q)$  è ben formata,  $k > 0$ ,  $vL(Q) = L(Q)$  )

int delastk(nodo\* & Q, int k, int y, int count)

POST=(se  $vL(Q)$  contiene  $m \geq 0$  nodi con  $\text{info}=y$  e  $m + \text{count} \geq k$ , allora vengono eliminati gli ultimi nodi di  $vL(Q)$  con  $\text{info}=y$ , fino al massimo di  $k$  (se  $m < k$  allora vengono eliminati  $m$  nodi con  $\text{info}=y$ ) e la funzione restituisce il numero di nodi che devono ancora essere eliminati per arrivare a  $k$ ) && (altrimenti,  $L(Q) = vL(Q)$ ) && (i nodi eliminati sono stati deallocati)

**Esempio 2:** Considerate i dati dell'Esempio 1. Quando delastk raggiunge il suffisso  $4 \rightarrow -1 \rightarrow 0$ , count deve essere 2 a indicare che prima di arrivare lì sono stati visti 2 nodi con  $\text{info}=-1$ , inoltre per  $4 \rightarrow -1 \rightarrow 0$ ,  $m=1$  e  $2+1 \geq k=2$ , quindi, al ritorno di delastk su  $4 \rightarrow -1 \rightarrow 0$ , la lista deve essere diventata  $4 \rightarrow 0$  e delastk deve restituire il valore 1 ad indicare che resta ancora da eliminare un nodo con  $\text{info}=-1$ .

**Correttezza:** dimostrare la correttezza della funzione delastk rispetto alle PRE e POST date.