

Esercizio 3 del 12/5/2015 da consegnare per il 19/5

E' un esercizio ricorsivo sulle liste concatenate. Si chiede di scrivere una funzione ricorsiva

nodo* del_z(nodo* n, int z)

che deve eliminare da lista(n) tutti i nodi che hanno campo info uguale a z e restituire con il return la lista dei nodi rimanenti. **Nessun nuovo nodo va creato da del_z.** La funzione restituisce una lista composta da nodi in lista(n) e dealloca (con delete) i nodi di lista(n) con campo info uguale a z.

La struttura nodo è la seguente:

```
struct nodo{int info; nodo* next; nodo(int a=0, nodo* b=0){info=a; next=b;}};
```

Esempio: sia lista(n)= 3->1->3->2->3->3 e z=3. I nodi da eliminare da lista(n) sono 4. Essi vanno deallocati. Quello che resta della lista originale è la lista 1->2 che va restituita col return.

Dovete scrivere il main che apre i file "input" e "output", legge un intero dim e poi dim interi con cui costruisce una lista concatenata lista(n) invocando la funzione crea dell'esercizio 1 di questa settimana e poi legge un ultimo intero z e quindi invoca del_z(n,z) e per finire stampa (ricorsivamente) la lista restituita da del_z.

Quindi nell'esempio precedente il programma deve stampare su "output": 1->2 seguito da "fine".

Consiglio: fare attenzione ai casi limite. Per esempio lista(n) vuota oppure z assente da lista(n), oppure lista(n) composta di tutti i nodi con campo info uguale a z.

Correttezza: scrivere pre e post-condizione di del_z e dimostrare che la funzione è corretta rispetto a questa coppia di asserzioni.