

## Scritto di programmazione del 30/6/2015

Data una lista concatenata C vogliamo eliminare da C quei nodi che sono preceduti (anche non immediatamente) da nodi con uguale campo info. La cosa è spiegata nel seguente esempio.

**Esempio:** se la lista è C= 2->0->1->1->0->3->2 allora vogliamo farla diventare 2->0->1->3 eliminando per esempio il quarto nodo perché il suo campo info=1 è uguale a quello del terzo nodo. Anche il quinto nodo va eliminato perché il suo campo info=0 è lo stesso di quello del secondo nodo e così via. Per un dato campo info che ripete, si vuole mantenere solo il primo dei nodi che lo contiene.

Per realizzare questo compito si chiede di realizzare 2 funzioni, una iterativa ed una ricorsiva.

**Funzione iterativa:** `nodo* togl(nodo*C, int y)` deve soddisfare le seguenti pre e post-condizioni:

PRE=(lista(C) è corretta e y è definito)

POST=(restituisce la lista che resta eliminando da C i nodi con campo info=y. I nodi tolti da C sono deallocati. Nessun nodo nuovo rispetto a C è allocato)

**Funzione ricorsiva:** `nodo* no_rip(nodo*C)` che soddisfa le seguenti pre e post-condizioni:

PRE=(lista(C) è corretta)

POST=(restituisce quello che resta di C mantenendo solo il primo nodo con un dato campo info ed eliminando i nodi successivi con lo stesso campo info. I nodi tolti da C sono deallocati. Nessun nuovo nodo rispetto a C è allocato)

L'idea è che la funzione `no_rip`, per fare quello che deve fare, invoca la funzione `togli`.

**Consiglio:** l'uso della seguente struttura FIFO (vista in classe) :

```
struct FIFO{nodo* primo,*fine; FIFO(nodo*a=0, nodo*b=0){primo=a; fine=b;}};
```

dove `primo` punta al primo nodo di una lista e `fine` punta all'ultimo nodo della stessa lista, permette di realizzare la funzione `togli` in modo semplice. La struttura dati FIFO si accompagna con la funzione `push_end` che permette di aggiungere un nuovo nodo in fondo alla lista gestita da un valore FIFO.

**Correttezza:** scrivere una dimostrazione induttiva della correttezza di `no_rip` rispetto alla PRE e POST date. La dimostrazione deve assumere che `togli` sia corretta rispetto alle sue PRE e POST.

**Esercizio finale:** Si chiede anche di scrivere una funzione iterativa che, date 2 liste concatenate T e C, elimina da T (deallocandoli) tutti quei nodi che hanno campo info uguale ad un nodo di C. La funzione deve restituire la nuova lista T. La funzione può usare una o entrambe le funzioni `no_rip` e `togli` e deve avere la seguente segnatura:

```
void elim(nodo*&T, nodo*C);
```