

## Esercizio 2 del 22/5/2015 da iniziare nel laboratorio del 22/5/2015

Si tratta di "sgranare" una lista concatenata L in 2 liste concatenate, L1 e L2, tali che L1 contenga i nodi di L in prima, terza, quinta, insomma nelle posizioni dispari di L, mentre L2 deve contenere i nodi di L nelle posizioni pari. I nodi di L1 ed L2 devono mantenere tra loro lo stesso ordine relativo che avevano in L. Nel seguito useremo la seguente terminologia per indicare le 2 liste L1 ed L2 ottenute sgranando L:  $SG(L)=(L1,L2)$ .

**Attenzione:** Nessun nuovo nodo va creato e nessun nodo va deallocato. Si usano solo i nodi di L per comporre L1 ed L2.

Vediamo un esempio.

**Esempio:** sia  $L=3 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 0 \rightarrow 0 \rightarrow 1$ , allora  $L1=3 \rightarrow 1 \rightarrow 0 \rightarrow 1$  e  $L2=2 \rightarrow 3 \rightarrow 0$ .

Si richiede di risolvere l'esercizio in 2 modi diversi:

a) il primo modo è attraverso una funzione ricorsiva sgrana che ha il seguente prototipo:

`dueL sgrana(nodo*L)`, dove `dueL` è il seguente tipo struttura che trovate già nel file `esercizio2_22_5_2015.cpp` che è dato:

```
struct dueL {nodo* prima, * seconda; dueL(nodo*a=0, nodo* b=0){prima=a; seconda=b;}};
```

come vedete un valore `dueL` ha i 2 campi `prima` e `seconda` capaci di contenere 2 liste concatenate. L'idea è che `sgrana` restituisca col `return` un valore `V` di tipo `dueL` tale che `V.prima=L1` e `V.seconda=L2`, dove  $SG(L)=(L1,L2)$ .

b) il secondo modo è attraverso una funzione col seguente prototipo: `nodo* sgrana2(nodo*&L)`, l'idea è quella di calcolare ancora L1 ed L2 di  $SG(L)$ , ma di restituire L1 con il `return` e L2 attraverso il parametro `L`, passato per riferimento.

E' dato un `main` che compie le letture, costruisce 2 liste su cui invocare `sgrana` e `sgrana2`, e fa le stampe dei risultati ottenuti da queste invocazioni. Esso invoca le 2 funzioni ricorsive `sgrana` e `sgrana2` che sono da fare.

**Correttezza:** scrivere pre- e post-condizioni delle 2 funzioni `sgrana` e `sgrana2` e dimostrare induttivamente la correttezza di `sgrana`. Nelle post potete usare la notazione `SG` introdotta prima.