

## Esame Scritto di Programmazione del 10/7/2018

Data una lista concatenata  $L$  ed un array  $A$  di interi maggiori o uguali a 0 e crescenti, si tratta di distribuire i nodi di  $L$  su due liste  $L1$  ed  $L2$ , dove  $L1$  consiste dei nodi corrispondenti agli indici contenuti in  $A$  e  $L2$  contiene gli altri nodi di  $L$  (cioè quelli il cui indice non appare in  $A$ ).

**Esempio:** Data la lista concatenata  $L = 3 \rightarrow 1 \rightarrow 2 \rightarrow 20 \rightarrow -11$ , assegniamo indici ai suoi nodi partendo da 0. Quindi il primo nodo (3) ha indice 0, il secondo (1) ha indice 1 e così via. Ora, se  $A = [2, 4, 10]$ ,  $L$  verrà scomposta nelle 2 liste  $L1$  e  $L2$  seguenti:  $L1 = 2 \rightarrow -11$  e  $L2 = 3 \rightarrow 1 \rightarrow 20$ . Si osservi che l'ultimo elemento di  $A$  non serve a nulla in quanto  $L$  contiene solo 5 nodi, pertanto va semplicemente ignorato.

Si richiede di realizzare questa operazione in modo iterativo e in modo ricorsivo. Più precisamente, si chiede di risolvere i seguenti esercizi:

### 1) Esercizio iterativo:

Realizzare una funzione iterativa `distr_it` che soddisfa le seguenti specifiche:

**PRE**=(lista( $L$ ) è ben formata,  $A$  contiene  $\text{dim}A$  elementi non negativi e crescenti,  $\text{dim}A > 0$ ,  $vL = \text{lista}(L)$ )

**nodo\* distr\_it(nodo\*&L, int\*A, int dimA)**

**POST**=(la funzione restituisce col return la lista  $L1$  dei nodi di  $vL$  i cui indici sono in  $A$  - nello stesso ordine relativo che avevano in  $vL$  - mentre  $L$  diventa  $L2$ , cioè la lista dei nodi di  $vL$  i cui indici non sono in  $A$  - nello stesso ordine relativo che avevano in  $vL$ )

*Consiglio:* conviene usare delle **variabili di tipo coda** (la cui interfaccia e implementazione è già inclusa nel programma dato), assieme alla funzione `push_end`.

### 2) Esercizio ricorsivo:

Realizzare una funzione ricorsiva `distr_ric` che soddisfa le seguenti specifiche:

**PRE**=(lista( $L$ ) è ben formata,  $A$  ha  $\text{dim}A \geq 0$ , elementi non negativi e crescenti e, se  $\text{dim}A > 0$ , allora  $n \leq A[0]$ ,  $vL = \text{lista}(L)$ )

**nodo\* distr\_ric(nodo\*&L, int\*A, int dimA, int n)**

**POST**=(restituisce col return la lista  $L1$  dei nodi di  $vL$  i cui indici, sommati a  $n$ , sono in  $A$  - nello stesso ordine relativo che avevano in  $vL$  - e  $L$  diventa  $L2$ , cioè la lista dei nodi di  $vL$  i cui indici sommati a  $n$  non sono in  $A$  - nello stesso ordine relativo che avevano in  $vL$ )

### 3) Correttezza:

- Scrivere l'invariante del ciclo principale di `distr_it`
- Scrivere la dimostrazione induttiva di `distr_ric`