

Laboratorio di Algoritmi e Strutture Dati

Prova di laboratorio, turno 1, 6 febbraio 2003

Si consideri il tipo `list` delle liste di interi realizzate con array, come definito nel file `lib.h`:

```
#define MAX 10

typedef struct list list;

struct list{
    int info[MAX];
    int lg;
};
```

dove il campo `lg` contiene la lunghezza della lista.

1. (**punti -2 ; 0**) Definire la funzione `bool no_rep(list l)` che restituisce `TRUE` se la lista `l` non contiene elementi ripetuti, `FALSE` altrimenti.
2. (**punti 0 ; 1**) Facendo uso delle due funzioni `del_rep` e `cat` definite in `lib.o` (vedere anche l'header corrispondente `lib.h`), definire la funzione `list merge(list l1, list l2)` che restituisce una nuova lista senza ripetizioni contenente tutti e soli gli elementi di `l1` e di `l2` in modo che:
 - l'ordine sia rispettato, ossia tutti gli elementi di `l1` devono precedere quelli di `l2` e gli elementi di `l1` ed `l2` devono essere nello stesso ordine;
 - per ogni elemento deve rimanere solo la sua prima occorrenza (a partire da `l1`).

Ad esempio se `l1` è 1 1 5 0 0 2 ed `l2` è 0 5 3 1 4, allora `merge` restituisce la lista 1 5 0 2 3 4.

Supporre che la lunghezza delle due liste non ecceda `MAX`; nel caso in cui il risultato dovesse avere lunghezza maggiore di `MAX`, la funzione deve restituire una lista lunga esattamente `MAX` ottenuta concatenando `l1` con i primi `MAX-del_rep(l1).lg` elementi non ripetuti di `l2`.

Nota bene: per ottenere una valutazione positiva è obbligatorio usare le due funzioni `del_rep` e `cat`.

3. (**punti 0 ; 2**) Definire la funzione `list my_cat(list l1, list l2)` che restituisce una nuova lista ottenuta concatenando `l1` con `l2`.

Ad esempio se `l1` è 1 2 3 2 ed `l2` è 2 1 0 0, allora `my_cat` restituisce la lista 1 2 3 2 2 1 0 0.

Supporre che la lunghezza delle due liste non ecceda `MAX`; nel caso in cui il risultato dovesse avere lunghezza maggiore di `MAX`, la funzione deve restituire una lista lunga esattamente `MAX` ottenuta concatenando `l1` con i primi `MAX-11.lg` elementi di `l2`.

Nota bene: è vietato usare la funzione `cat` definita in `lib.o`.

4. (**punti 0 ; 2**) Definire la funzione `list my_del_rep(list l)` che restituisce una nuova lista ottenuta cancellando da `l` tutti gli elementi ripetuti in modo che ne rimanga solo la *prima* occorrenza.

Ad esempio, se `my_del_rep` viene chiamata sulla lista 1 1 2 1 2 3 4 3 1 4 3 1 2 allora deve restituire la lista 1 2 3 4.

Nota bene: è vietato usare la funzione `del_rep` definita in `lib.o`.