

Laboratorio di Linguaggi di Algoritmi e Strutture Dati

Prova di laboratorio, turno 3, 19 giugno 2002

Si consideri il tipo `list` delle liste di interi realizzate con puntatori semplici senza elemento fittizio (detto anche header), come definito nel file `lib.h`:

```
typedef struct cell *list;

struct cell{
    int info;
    list next;
};
```

1. (**punti 0**) Si definisca in modo ricorsivo la funzione `bool is_rep(list l)` che restituisce `TRUE` se la lista `l` contiene elementi contigui uguali, `FALSE` altrimenti.
2. (**punti 1**) Si definisca la funzione `int_pair get(list l)` che restituisce la coppia dei primi due elementi della lista `l` (a partire dalla testa) diversi tra loro; in caso tali elementi non esistano restituisce la coppia `(0,0)`. Il tipo `int_pair` è definito nel file `lib.h`.
3. (**punti 2**) Si definisca in modo ricorsivo la funzione `bool no_rep(list l)` che restituisce `TRUE` se e solo se la lista `l` non ha elementi ripetuti.

La funzione deve utilizzare le seguenti funzioni definite in `ese3.h` e `ese3.o`:

- `bool is_rep(list l)`, con lo stesso comportamento della funzione al punto 1.
- `int_pair get(list l)`, con lo stesso comportamento della funzione al punto 2.
- `list_pair split(list l,int i)` restituisce una coppia di liste, la prima contenente tutti e soli gli elementi `j` di `l` tali che $j \leq i$, la seconda tutti e soli gli elementi `j` di `l` tali che $j > i$ (vedere anche il punto 4). Il tipo `list_pair` è definito nel file `lib.h`.

Lo schema da seguire è il seguente:

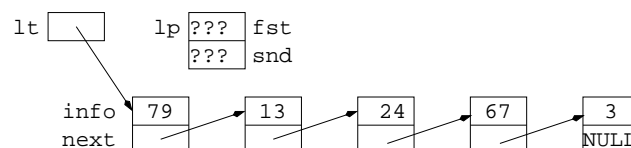
- base: ...
 - passo induttivo: controllare se la lista ha elementi contigui uguali; se sì allora ..., altrimenti ottenere i primi due elementi della lista diversi tra loro, calcolarne il minimo e usare tale valore per dividere in due la lista (con la funzione `split`).
- Infine, richiamare ricorsivamente `no_rep` sulle due sotto-liste ...

Nota bene: qualsiasi altro tipo di algoritmo, anche se corretto, verrà valutato 0 punti!

La funzione deve esclusivamente utilizzare le celle della lista `l` senza allocarne delle nuove.

4. (**punti 2**) Si definisca la funzione `list_pair split(list l,int i)` utilizzata al punto 3. Come già specificato, la funzione restituisce una coppia di liste, la prima contenente tutti e soli gli elementi `j` di `l` tali che $j \leq i$, la seconda tutti e soli gli elementi `j` di `l` tali che $j > i$.

A titolo di esempio, si consideri la variabile `lt` contenente la seguente lista:



Dopo l'istruzione `lp=split(lt,13)`, si ottiene la seguente configurazione:

