

---

## Separació dels elements de les posicions parells i senars d'una llista simplement encadenada

---

X66872\_ca

---

Donada la classe *Llista* que permet guardar seqüències d'enters amb una llista simplement encadenada, sense fantasma i no circular, cal implementar el mètode

```
void separa( Llista &l2)
```

que a partir d'una llista *l2* buida, separa els elements del paràmetre implícit quedant al paràmetre implícit els elements de les posicions senars i a *l2* els de les posicions parells.

Cal enviar a jutge.org només la implementació del mètode *separa*. La classe *Llista* té la següent especificació:

```
#include <vector>
#include <cstddef>
using namespace std;
typedef unsigned int nat;

class Llista {
    // Llista simplement encadenada, sense fantasma i no circular.
private:
    struct node {
        int info; // Informació del node
        node *seg; // Punter al següent element
    };
    node *_prim; // Punter al primer element
    nat _long; // Nombre d'elements

public:
    Llista ();
    // Pre: True
    // Post: El p.i. és una llista buida.

    Llista (const vector<int> &v);
    // Pre: True
    // Post: El p.i. conté els elements de v amb el mateix ordre.

    ~Llista ();
    // Post: Destruïx els elements del p.i.

    nat longitud() const;
    // Pre: True
    // Post: Retorna el nombre d'elements del p.i.

    void mostra() const;
    // Pre: True
    // Post: Mostra el p.i. pel canal estàndard de sortida.
```

```

void separa ( Llista &l2);
// Pre: l2 és buida
// Post: S'han separat els elements del p.i., quedant al p.i. els elements
// de les posicions senars i a l2 els de les posicions parells.
// Exemple: [2 5 3 8 1] quedaria [2 3 1] i l2 = [5 8]
};

```

Per testejar la solució, jutge.org ja té implementats la resta de mètodes de la classe *Llista* i un programa principal que processa línies d'enters amb els que crea llistes i després crida el mètode *separa*.

## Entrada

L'entrada conté diverses línies formades per seqüències d'enters. Cadascuna d'elles són els elements que tindrà cada llista.

## Sortida

Per a cada línia d'entrada, escriu una línia amb el resultat després d'haver separat els elements de les posicions senars i parells de la llista: Per cada llista mostra el nombre d'elements de la llista seguit d'un espai i dels elements de la llista entre claudàtors i separats per espais.

## Observació

Cal enviar la solució (el fitxer *solution.cpp*) comprimida en un fitxer *.tar*:

```
tar cvf solution.tar solution.cpp
```

Només cal enviar la implementació del mètode *separa*. Seguiu estrictament la definició de la classe de l'enunciat.

### Exemple d'entrada

```

3 -6 8 0 4 -2
5

9 7
1 2 3

```

### Exemple de sortida

```

3 [3 8 4] 3 [-6 0 -2]
1 [5] 0 []
0 [] 0 []
1 [9] 1 [7]
2 [1 3] 1 [2]

```

## Informació del problema

Autor : Jordi Esteve

Generació : 2020-10-08 23:40:20

© Jutge.org, 2006–2020.

<https://jutge.org>