

---

## Duplicació dels elements d'una llista simplement encadenada amb fantasma X18740\_ca

---

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

```
void duplica ()
```

que duplica els elements del paràmetre implícit.

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

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

class Llista {
// Llista simplement encadenada, amb fantasma i no circular.
private:
    struct node {
        int info; // Informació del node
        node *seg; // Punter al següent element
    };
    node *_prim; // Punter a l'element fantasma
    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 duplica ();
// Pre: True
```

```
// Post: S'han duplicat els elements del p.i.  
// Exemple: [2 5 3] quedaria [2 2 5 5 3 3]  
};
```

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 `duplica`.

## 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 duplicat els elements de la llista: 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 `duplica`. Seguiu estrictament la definició de la classe de l'enunciat.

## Exemple d'entrada

```
3 -6 8 0 4 -2  
5  
  
9 7
```

## Exemple de sortida

```
12 [3 3 -6 -6 8 8 0 0 4 4 -2 -2]  
2 [5 5]  
0 []  
4 [9 9 7 7]
```

## Informació del problema

Autor : Jordi Esteve  
Generació : 2020-10-21 21:17:10

© `Jutge.org`, 2006–2020.  
<https://jutge.org>