

---

## Duplicació dels positius i eliminació dels negatius d'una llista simplement encadenada X41943\_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 duplica_positius_elimina_negatius ()
```

que duplica els elements positius i elimina els elements negatius del paràmetre implícit.

Cal enviar a jutge.org només la implementació del mètode *duplica\_positius\_elimina\_negatius*.

La classe *Llista* té la següent especificació:

```
#include <vector>
#include <cstdlib>
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 duplica_positius_elimina_negatius ();
```

```

// Pre: True
// Post: S'han duplicat els elements positius i
// s'han eliminat els elements negatius del p.i.
// Exemple: [0 3 -6 8 0 4 -2 0] =¿ [0 3 3 8 8 0 4 4 0]
};

```

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\_positius\_elimina\_negatius*.

## 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 positius i eliminat els elements negatius 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\_positius\_elimina\_negatius*. Segueix estrictament la definició de la classe de l'enunciat.

### Exemple d'entrada 1

```
0 3 -6 8 0 4 -2 0
```

### Exemple d'entrada 2

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

### Exemple d'entrada 3

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

### Exemple d'entrada 4

### Exemple d'entrada 5

```
0
3
-3
```

### Exemple de sortida 1

```
9 [0 3 3 8 8 0 4 4 0]
```

### Exemple de sortida 2

```
9 [3 3 8 8 0 4 4 5 5]
9 [3 3 6 6 0 2 2 5 5]
```

### Exemple de sortida 3

```
9 [6 6 8 8 0 4 4 2 2]
5 [8 8 0 4 4]
```

### Exemple de sortida 4

```
0 []
```

### Exemple de sortida 5

```
1 [0]
2 [3 3]
0 []
```

### Exemple d'entrada 6

```
0 0
-3 -5
3 5
-3 5
3 -5
3 0
0 3
-5 0
0 -5
```

### Exemple de sortida 6

```
2 [0 0]
0 []
4 [3 3 5 5]
2 [5 5]
2 [3 3]
3 [3 3 0]
3 [0 3 3]
1 [0]
1 [0]
```

### Informació del problema

Autor : Jordi Esteve

Generació : 2023-01-26 18:28:54

© *Jutge.org*, 2006–2023.

<https://jutge.org>