
Distribuir els elements d'una llista en dues llistes**X64830_ca**

Escriviu una funció `separa` que, donada una llista `lp` de punts i un real `x`, torni dues llistes: `lp1` amb els punts de `lp` amb la coordenada `x` estrictament menor que `x`, i `lp2` amb els punts de `lp` amb la coordenada `x` estrictament major que `x`. `lp` ha de quedar buida. Observació: Si algun punt a la llista `lp` té la coordenada `x` igual a `x`, no apareixerà a cap de les llistes de sortida.

```
void separa(list<Punt> &lp, float x, list<Punt> &lp1, list<Punt> &lp2);  
/* Pre: lp1 i lp2 estan buides */  
/* Post:  
    lp1 conté els punts de lp amb la coordenada x estrictament menor que x,  
    lp2 conté els punts de lp amb la coordenada x estrictament major que x,  
    lp està buida.*/
```

Entrada

Com a entrada hi haurà el nombre de punts i els punts de la llista (cada punt té dues coordenades reals). A continuació hi haurà varis valors de la coordenada `x`.

Sortida

Com a sortida es mostrarà l'estructura de la llista de punts original i, per cada coordenada `x` d'entrada, el valor `x` i les llistes `lp1` i `lp2` obtingudes al separar la llista original amb el valor `x`.

Observació

A més de la funció `separa` dins del fitxer `list_punt_separa.cpp`, heu d'implementar els fitxers `listIOpunt.hpp` i `listIOpunt.cpp`, amb les operacions de lectura i escriptura per a llistes instanciades amb `Punt`. Podeu agafar com a model els fitxers `listIOint.hpp` i `listIOint.cpp` que contenen les operacions de lectura i escriptura per a llistes d'enters. Heu d'enviar la solució comprimida en un fitxer `.tar`:

```
tar cvf program.tar list_punt_separa.cpp listIOpunt.hpp listIOpunt.cpp
```

Observeu que per compilar us donem el `Makefile`, el mòdul `Punt`, la capçalera del mòdul funcional `list_punt_separa.hpp` i el programa principal `program.cpp`.

Exemple d'entrada 1

```
5  
2 1  
6 5  
7 0  
3 2  
5 0  
3  
2  
7
```

Exemple de sortida 1

```
[(2, 1), (6, 5), (7, 0), (3, 2), (5, 0)]  
3  
[(2, 1)]  
[(6, 5), (7, 0), (5, 0)]  
2  
[]  
[(6, 5), (7, 0), (3, 2), (5, 0)]  
7  
[(2, 1), (6, 5), (3, 2), (5, 0)]  
[]
```

Informació del problema

Autoria: Neus Català - Jordi Esteve

Generació: 2026-01-25T21:20:22.465Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>