
Trams creixents d'una llista**X97020_ca**

Considerem un tipus T amb una relació d'ordre, per tant amb operacions $==, <, >, \geq, \leq$.

Un tram d'una llista d'elements de tipus T és una subseqüència màxima tal que cada element és més gran o igual que l'anterior.

Afegiu una operació pública a la classe *Llista* vista a teoria tal que, donada una llista com a paràmetre implícit i un vector de llistes buit v , compleixi la següent especificació

```
void trams(vector<Llista<T>>& v)
/* Pre: v.size() = 0, p.i. = L */
/* Post: el p.i. és buit, la mida de v és el nombre de trams de L,
        cada posició v[i] conté una llista amb l'i-èsim tram de L i el punt d'interès
```

Per exemple, si la llista és $\{3\ 7\ 10\ 10\ 4\ 6\ 6\ 12\ 6\ -2\ 5\}$, el vector ha d'acabar sent $v[0] = \{3\ 7\ 10\}$, $v[1] = \{4\ 6\ 6\ 12\}$, $v[2] = \{6\}$, $v[3] = \{2\ 5\}$.

Observació

No utilitzeu cap de les operacions primitives de les llistes, accediu directament als atributs de la classe *Llista*. La vostra solució no pot crear mai cap node nou, només pot modificar els apuntadors dels nodes ja existents.

Només s'ha d'enviar un fitxer anomenat `program.hh`, que no ha de fer cap *cout*, que contingui la funció amb la capçalera de l'enunciat i qualsevol altra funció auxiliar que cregueu convenient, sense la funció *main* i sense posar-hi cap *include*. A l'apartat *Public files* trobareu els fitxers que us calen per construir la vostra solució.

Per provar la vostra solució us recomanem crear a part un `main.cc`, que haurà de començar amb

```
#include "utils.PRO2"
#include <vector>
#include "LlistaIOint.hh"
```

Informació del problema

Autoria: Borja Valles (responsable)

Generació: 2026-01-27T18:57:48.894Z

© Jutge.org, 2006–2026.

<https://jutge.org>