
Mètode de la classe Llista per girar cap al sentit contrari del rellotge tantes vegades com indiqui n X24601_ca

Implementeu un nou mètode de la classe List que mogui els elements de la llista cap al sentit contrari del rellotge tantes vegades com indiqui **n**. És a dir, el primer element passarà a ser l'últim, el segon passarà a ser el primer i així successivament **NOMÉS** tantes vegades com indiqui **n**.

D'entre els fitxers que s'adjunten en aquest exercici, trobareu `listCounterclockwise.old.hpp`, a on hi ha una implementació de la classe genèrica List. En primer lloc, haureu de fer:

```
cp listCounterclockwise.old.hpp listCounterclockwise.hpp
```

A continuació, haureu de buscar dins `listCounterclockwise.hpp` la part:

```
// Pre: 'n' indica la quantitat de posicions que cal desplaçar-se.  
// Post: Retorna una llista de string on el primer element de la llista passarà  
// a ser l'últim de la llista i el segon passarà a ser el primer element  
// de la llista, es repetirà tantes vegades com indiqui 'n'. En resum, moure  
// tants elements de la llista com indiqui 'n' cap al sentit contrari  
// del rellotge.  
// Descomenteu les següents dues línies i implementeu la funció:  
// void listCounterclockwise(int n) {  
// }
```

Haureu de descomentar les dues línies que s'indiquen i implementar aquest mètode. No toqueu la resta de la implementació de la classe, excepte si, per algun motiu, considereu que necessiteu afegir algun mètode auxiliar a la part privada.

La implementació d'aquest mètode hauria de consistir en modificar punters. De fet, possiblement qualsevol implementació produirà error d'execució.

D'entre els fitxers que s'adjunten a l'exercici també hi ha `program.cpp` (programa principal) i `Makefile` per a compilar. Per a pujar la vostra solució, heu de crear el fitxer `solution.tar` així:

```
tar cf solution.tar listCounterclockwise.hpp
```

Entrada

La entrada del programa és una seqüència d'instruccions del següent tipus que s'aniran aplicant sobre una llista que se suposa inicialment buida i un iterador que se suposa situat inicialment al principi (i final) d'aquesta llista:

```
push_front s (s és un string)  
push_back s (s és un string)  
pop_front  
pop_back  
it++  
it--  
*it  
listCounterclockwise 3
```

Se suposa que la seqüència d'entrada serà correcta (sense pop_front ni pop_back sobre llista buida, ni *it tenint it situat al end de la llista). Tampoc hi haurà pop_front just quan l'iterador estigui apuntant al primer element de la llista, ni hi haurà pop_back just quan l'iterador estigui apuntant a l'últim element de la llista (tingueu en compte que l'últim element de la llista no és el end de la llista).

El programa principal que us oferim ja s'encarrega de llegir aquestes entrades i fer les crides als corresponents mètodes de la classe list. Només cal que implementeu els mètodes abans esmentats.

Sortida

Per a cada instrucció *it, s'escriurà el contingut apuntat per l'iterador i mostrarà la llista de string. El programa que us oferim ja fa això. Només cal que implementeu el mètode abans esmentat.

Exemple d'entrada 1

```
listCounterclockwise 1
insert a
listCounterclockwise 1
it--
*it
push_back b
listCounterclockwise 1
it--
*it
push_back c
listCounterclockwise 2
it++
*it
push_back d
listCounterclockwise 2
it++
*it
push_back e
listCounterclockwise 3
it++
*it
```

Exemple d'entrada 2

```
insert n
listCounterclockwise 7
it--
push_front r
*it
push_front h
*it
push_front o
listCounterclockwise 7
push_back d
listCounterclockwise 8
it--
push_front j
listCounterclockwise 3
push_front s
it--
push_front b
```

Exemple de sortida 1

```
a a
b ba
a cba
d adcb
c beadc
```

```
push_back r
*it
push_back c
listCounterclockwise 5
push_front p
*it
push_back l
*it
push_back p
*it
push_front o
listCounterclockwise 6
push_back h
pop_front
push_back e
*it
*it
```

Exemple de sortida 2

```
n rn
n hrn
r bsnohjdr
```

```
r pjrdrbcsnoh
r pjrdrbcsnohl
r pjrdrbcsnohlp
r bsnohlpopjdrhe
r bsnohlpopjdrhe
```

Observació

Avaluació sobre 10 punts:

- Solució lenta: 6 punts.
- solució ràpida: 10 punts.

Informació del problema

Autoria: STUDENTS PRO1

Generació: 2026-01-25T21:04:54.865Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>