

---

## Distribució justa de cues

X13425\_ca

---

Donada una cua de parells d'enters, on el primer és un identificador d'usuari i el segon el temps esperat de la gestió que vol fer, dissenyeu una operació que reparteixi els usuaris en dues noves cues de tal manera que

- Cap persona esperi més a la nova cua que una que tenia al darrera inicialment.
- Totes esperin el mínim temps possible.
- Si un usuari pot anar a les dues cues, s'escollira la primera.

### Entrada

Una seqüència de parells d'enters acabada en 0 0.

### Sortida

El contingut de les dues cues.

### Observació

Heu d'enviar tres fitxers en un sol .tar:

- CuaIOParInt.hh amb les funcions

```
void llegirCuaParInt(queue<ParInt>& c);
// Pre: c és buida; el canal estandar d'entrada conté un nombre
// parell d'enters, acabat pel parell 0 0
// Post: s'han encuat a c els elements llegits fins al 0 0 (no inclòs)

void escriureCuaParInt(queue<ParInt> c);
// Pre: cert
// Post: s'han escrit al canal estandar de sortida els elements
de c
```
- CuaIOParInt.cc amb la seva codificació.
- program.cc amb el programa. La distribució d'una cua s'ha de programar en una operació a part. El main del programa només ha de llegir les cues, cridar l'esmentada operació i escriure els resultats.

Observeu que per als parells d'enters us donem la classe ParInt que detecta si el parell llegit és 0 0 i per compilar us donem el Makefile.

Aquest mateix exercici admet una segona solució on l'operació que fa la distribució només produeix una cua nova, tot traient elements de l'original, i escrivint les dues.

### Exemple d'entrada

3 2  
6 3

2 5  
11 1  
8 4

```
5 3
9 2
1 3
7 4
15 2
4 3
0 0
```

### Exemple de sortida

```
3 2
2 5
5 3
1 3
15 2

6 3
11 1
8 4
9 2
7 4
4 3
```

### Informació del problema

Autor : Xavier Messeguer (adaptador) Borja Valles (responsable)

Generació : 2020-10-06 17:50:19

© *Jutge.org*, 2006–2020.

<https://jutge.org>