

---

## Elimina punts d'una cua de punts

X36225\_ca

---

Escriviu el codi d'una funció **recursiva** `elimina_punts` que donada una cua de punts `c` i un punt `p`, retorni la cua resultant d'eliminar totes les aparicions del punt `p` de la cua `c`. Si la cua no conté el punt `p`, la cua no s'ha de modificar.

```
queue<Punt> elimina_punts(queue<Punt> c, Punt p);  
/* Pre: c = C i p = P */  
/* Post: retorna la cua C on s'han eliminat totes les aparicions del punt P */
```

### Entrada

Com a entrada hi haurà una cua: el nombre de punts i els punts que la formen. A continuació hi hauran un o més punts addicionals.

Per llegir la cua s'ha utilitzat l'operador `>>` que es troba definit en el mòdul `queueIOpunt`.

### Sortida

Com a sortida es mostrarà la cua original. A continuació es mostrarà una cua per cada punt d'entrada addicional que serà la cua obtinguda a l'eliminar aquest punt en la cua original.

Per escriure les cues s'ha utilitzat l'operador `<<` que es troba definit en el mòdul `queueIOpunt`.

### Observació

Heu d'enviar la solució comprimida en un fitxer `.tar`:

```
tar cvf program.tar cua_elimina.cpp
```

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

Jutge.org també us donarà un semàfor verd si envieu una solució iterativa, però no serà correcte doncs l'enunciat del problema demana que la solució enviada sigui **recursiva**.

### Exemple d'entrada 1

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

### Exemple de sortida 1

```
<(2, 1)|(6, 5)|(7, 0)|(6, 5)|(5, 0)|<
<(2, 1)|(7, 0)|(5, 0)|<
<(2, 1)|(6, 5)|(6, 5)|(5, 0)|<
<(2, 1)|(6, 5)|(7, 0)|(6, 5)|(5, 0)|<
```

### Informació del problema

Autoria: Neus Català - Jordi Esteve

Generació: 2026-01-25T21:09:56.787Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>