

---

## Control PRO2 - Torn 1 (primavera 2016)

X64632\_ca

---

Hem decidit implementar una classe `Cua_estudiants` usant un vector d'estudiants amb capacitat màxima. Les operacions de la classe són les habituals de les cues amb una nova funcionalitat: obtenir la nota màxima dels estudiants de la cua. A efectes d'aquesta funcionalitat, un estudiant sense nota es considerarà equivalent a un estudiant amb nota zero. També s'ha afegit una operació pública `full` que indica si la cua està plena. La nova funcionalitat està associada a una operació pública amb la següent especificació:

```
double nota_maxima() const;
/* Pre: el paràmetre implícit no està buit */
/* Post: el resultat és la nota màxima dels estudiants del paràmetre implícit
        (zero si cap d'ells té nota) */
```

Això ha suposat afegir un atribut especial a la classe, a més dels necessaris per a la implementació de les operacions habituals de les cues. Tenint això en compte, només heu d'implementar eficientment les següents operacions:

```
void push(const Estudiant &est);
/* Pre: el paràmetre implícit no està ple */
/* Post: el paràmetre implícit és com el paràmetre implícit original amb
        est afegit com a darrer element */

void pop();
/* Pre: el paràmetre implícit no està buit */
/* Post: el paràmetre implícit és com el paràmetre implícit original però
        sense el primer element afegit al paràmetre implícit original */

void escriure() const;
/* Pre: cert */
/* Post: s'han escrit pel canal estàndar de sortida els estudiants del
        paràmetre implícit per ordre d'arribada a la cua (del primer al darrer) */
```

### Observació

Heu de lliurar un fitxer `solucio.cc` amb una implementació eficient de les operacions `push`, `pop` i `escriure` que ha de tenir el següent format:

```
#include "Cua_estudiants.hh"

void Cua_estudiants::push(const Estudiant &est)
{
    ... // codi de la implementació
}

void Cua_estudiants::pop()
{
    ... // codi de la implementació
}

void Cua_estudiants::escriure() const
{
    ... // codi de la implementació
}
```

Copieu aquesta plantilla en el vostre `solucio.cc` i completeu-la. El vostre `solucio.cc` no pot contenir la implementació d'altres operacions de la classe.

A l'apartat *Public files* del Jutge us proveïm amb material addicional comprimit en un fitxer `.tar`. Podeu descomprimir aquest fitxer amb la comanda

```
tar -xvf nom_fitxer.tar
```

Aquest material addicional consisteix en els següents fitxers:

- `Cua_estudiants.hh`: l'especificació Pre/Post de totes les operacions públiques de la classe `Cua_estudiants`, així como la definició dels camps privats. Fixeu-vos que hi ha un atribut double `notamax`, que conté la nota màxima dels estudiants de la cua o zero si la cua és buida o cap estudiant té nota. Aquest atribut ens ha permès implementar amb la màxima eficiència l'operació `nota_màxima` i cal actualitzar-lo en les operacions modificadores de la classe. **És molt important que la implementació de les operacions que us hem encarregat tingui en compte i preservi l'invariant de la representació de la classe** `Cua_estudiants`. No podeu afegir noves operacions públiques o privades a la classe ni modificar els seus atributs.
- `Cua_estudiants.cc`: la implementació de totes de les operacions de la classe `Cua_estudiants` tret de les operacions `push`, `pop` i `escriure`.
- `Estudiant.hh`: l'especificació de la classe `Estudiant` i la definició dels seus atributs. No presenta cap novetat.
- `Estudiant.cc`: la implementació dels mètodes de la classe `Estudiant`.
- `pro2.cc`: un programa principal que podeu fer servir per provar les operacions públiques de la classe `Cua_estudiants`.
- `llegeixme.txt`: instruccions per a generar l'executable del programa `pro2` i provar-lo.

Valorarem positivament que la solució no contingui instruccions (especialment bucles o crides a operacions costoses) ni objectes (especialment vectors) innecessaris. No es pot emprar cap estructura de dades que no hagi aparegut a les sessions 1-4 de laboratori.

Quan feu els enviaments el Jutge us indicarà quants jocs de proves passeu i de quin tipus (públic o privat). El joc de proves anomenat `public` s'explica al fitxer `llegeixme.txt`. Els jocs de proves `privat1`, `privat2` i `privat3` proven situacions especials.

## Informació del problema

Autor : Professors de PRO2

Generació : 2016-03-14 18:23:55

© *Jutge.org*, 2006–2016.

<http://www.jutge.org>