
Control PRO2 - Problema 2 (Tardor 2019)

X74199_ca

Es disposa de $0 \leq k$ beques que s'atorgaran als k estudiants amb millors notes d'un conjunt. Si hi ha més d'un estudiant amb la mateixa nota, es prioritzaran per a les beques els estudiants amb DNI més gran. En general, si un estudiant A està en millors condicions que un altre B per obtenir beca, direm que A és "millor" o "més prioritari" que B .

A més, tots els estudiants becats han de tenir una nota superior o igual a un cert valor donat m , amb $0 \leq m \leq \text{Estudiant::nota_maxima}()$; si el nombre d'estudiants que satisfan aquesta condició és més petit que k , llavors les beques corresponents es quedaran sense atorgar.

S'han adaptat les classes `Cjt_estudiants` i `Estudiant`, que heu vist al laboratori, per donar suport a la gestió de les beques (**fixeu-vos a les noves operacions i camps**). Noteu també que si volem llegir un conjunt, hem de proporcionar primer de tot els valors k i m .

En aquest exercici heu d'implementar dues de les operacions de la classe `Cjt_estudiants`, concretament

```
void afegir_estudiant(const Estudiant &est, bool& b);
/* Pre: el paràmetre implícit no està ple */
/* Post: b = indica si el p.i. original conté un estudiant amb el dni d'est;
    si b = fals, s'ha afegit l'estudiant est al paràmetre implícit */

int immediatament_millor(int i) const
/* Pre: 0<=i<nest; vest[i] te nota */
/* Post: el resultat és la posició en vest[0..nest-1] de l'estudiant
    immediatament millor que vest[i], si n'hi ha; -1 en cas contrari */
```

Observació

Heu de lliurar un fitxer `solution.cc` amb una implementació eficient de les operacions `afegir_estudiant` i `immediatament_millor`. Aquest fitxer no pot contenir la implementació d'altres operacions de la classe.

A l'apartat *Public files* del Jutge (icona del gatet) us proveïm amb una plantilla pel fitxer `solution.cc` que només cal completar. A més, hi trobareu material addicional, tot 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:

- `Cjt_estudiants.hh`: les capçaleres i l'especificació Pre/Post de les operacions públiques i privades de la classe `Cjt_estudiants`, així com la definició dels seus atributs.
- `Cjt_estudiants.cc`: la implementació de totes de les operacions de la nova versió de la classe `Cjt_estudiants` tret de les operacions `afegir_estudiant` i `immediatament_millor`.
- `Estudiant.hh`: l'especificació de la classe `Estudiant` i la definició dels seus atributs.
- `Estudiant.cc`: la implementació dels mètodes de la classe `Estudiant`.
- `pro2.cc`: un programa principal que podeu fer servir per provar la solució d'aquest exercici.
- `llegeixme.txt`: instruccions per a generar i provar l'executable del programa `pro2`.

- `sample.inp`: entrada del joc de proves públic
- `sample.cor`: sortida del joc de proves públic
- `solution.cc`: plantilla explicada previament

És fonamental que la solució sigui eficient en temps i espai. En particular, s'han d'evitar instruccions innecessàries (especialment bucles o crides a operacions costoses) i no es poden fer servir objectes auxiliars de les classes `vector` o `Cjt_estudiants`. 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`. El `privat1` és molt semblant al `public`. Els `privat2` i `privat3` proven situacions especials.

Informació del problema

Autoria: Professors de PRO2

Generació: 2026-01-25T21:24:57.288Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>