

Aquest és un problema del jutge per fer lliuraments de l'examen de la pràctica

- **Aquest examen dura una hora i quaranta-cinc minuts**
- **No es corregirà cap lliurament que no compili**
- L'examen es fa sense apunts
- El 50% de la nota és de l'execució, l'altre 50% de la correcció manual
- Es corregirà el darrer enviament amb més jocs de proves correctes
- En cas de no passar cap joc de proves es farà la correcció manual del darrer enviament que compili
- El nombre d'enviaments no fa baixar la nota

Descarregueu els fitxers públics per obtenir el material:

- enunciat detallat d'aquest examen
- enunciat de la pràctica
- especificació de `BinTree`
- especificació de `list`
- fitxer `llegeixme.txt`
- codi ja implementat
- fitxer `Makefile`
- joc de proves públic

**Després de llegir l'enunciat de l'examen atentament**, així com el fitxer `llegeixme.txt`, apliqueu la instrucció `make plantillas`. Us apareixeran els fitxers `priv_Cluster.hh` i `solution.cc` que haureu fer servir de plantilla. A més, `solution.cc` conté l'especificació de les operacions públiques que heu d'implementar. Les operacions privades que necessiteu, que només podran ser de la classe `Cluster` les decideu vosaltres: heu d'escriure la capçalera a `priv_Cluster.hh` i la implementació a `solution.cc`.

Les operacions que heu d'implementar, pertanyen a les classes `Cluster` i `Pendientes`. Ara us resumim les novetats més importants de la implementació que us donem d'aquestes classes.

Els atributs d'un objecte `Cluster` són:

```
BinTree<int> arg;  
vector<Procesador> chips;  
int n_procesador;
```

i el seu invariant de representació és:

```
chips.size() = n_procesador
arq té n_procesador elements, que són una permutació de [1..n_procesador]
```

Els atributs d'un objecte `Pendientes` són:

```
list <Proceso> pro_pen;
```

i el seu invariant de representació és:

```
els processos de pro_pen estan ordenats decreixentment per temps
i, en cas d'empat, creixentment per id
```

Haureu de fer servir una operació ja implementada de `Cluster`

```
void poner_proceso_en_procesador(int id_procesador, const Proceso & p, bool & puesto);
/* PRE : 1 <= id_procesador <= numero de elementos del p.i.; no hay ningún
proceso en el procesador id_procesador del p.i. con el mismo id que p */
/* POST : si p cabe en el procesador id_procesador del p.i., pasa a
ejecutarse en el dicho procesador y "puesto" es cierto, en caso
contrario el proceso no se ejecuta en el procesador y "puesto" es
falso */
```

així com diverses operacions de `Procesador` i `Proceso` que trobareu especificades als corresponents fitxers `.hh`.

## Entrada

Una seqüència d'instruccions seguint el format de l'enunciat de l'examen i del joc de proves públic.

## Sortida

El seu resultat seguint el format de l'enunciat de l'examen i del joc de proves públic.

## Observació

El Jutge prova el vostre lliurament mitjançant 4 jocs de proves:

- `sample`: el joc de proves públic.
- `privat1`: joc de proves similar al públic.
- `privat2`: joc de proves privat que fa èmfasi en afegir processos pendents.
- `privat3`: joc de proves privat que fa èmfasi en afegir un procés al cluster.

Heu de lliurar un fitxer `solution.tar` creat amb la instrucció `make tar` definida al `Makefile` amb una implementació eficient de les operacions que es demanen.

## Exemple d'entrada

```
configurar_cluster 10
5 3 2 0 8 0 0 0 1 9 0 0 7 4 6 0 0 0 10 0
10
20
30
40
50
10
20
30
40
50
10
agregar_proceso_pendiente 100 10 10
enviar_procesos_a_cluster 10
consultar_procesador 1
consultar_procesador 5
consultar_procesador 9
```

```

consultar_procesador 10
agregar_proceso_pendiente 101 41 20
agregar_proceso_pendiente 102 1 1
consultar_pendientes
enviar_procesos_a_cluster 2
consultar_procesador 5
consultar_procesador 10
agregar_proceso_pendiente 103 60 8
agregar_proceso_pendiente 104 12 100
agregar_proceso_pendiente 105 20 80
agregar_proceso_pendiente 106 7 10
agregar_proceso_pendiente 107 30 20
agregar_proceso_pendiente 108 15 40
agregar_proceso_pendiente 109 30 20
agregar_proceso_pendiente 110 15 40
consultar_pendientes
enviar_procesos_a_cluster 6
consultar_pendientes
consultar_procesador 1
consultar_procesador 2
consultar_procesador 3
consultar_procesador 4
consultar_procesador 5
consultar_procesador 6
consultar_procesador 7
consultar_procesador 8
consultar_procesador 9
consultar_procesador 10
avanzar_tiempo 11
consultar_procesador 1
consultar_procesador 2
consultar_procesador 3
consultar_procesador 4
consultar_procesador 5
consultar_procesador 6
consultar_procesador 7
consultar_procesador 8
consultar_procesador 9
consultar_procesador 10
consultar_pendientes
poner_proceso_en_procesador 6 111 1000 10
consultar_procesador 6
poner_proceso_en_procesador 6 112 10 10
consultar_procesador 6
quitar_proceso_de_procesador 4 200
consultar_procesador 4
quitar_proceso_de_procesador 4 107
consultar_procesador 4
configurar_cluster 2
1 2 0 0 0
100
200
enviar_procesos_a_cluster 100
consultar_procesador 1
consultar_procesador 2
acabar

```

## Exemple de sortida

```

Procesador 1
Procesador 5
  100 10 10
Procesador 9
Procesador 10
Procesos pendientes
  101 102
Procesador 5
  100 10 10
  102 1 1
Procesador 10
  101 41 20
Procesos pendientes
  104 105 108 110
  107 109 106 103
Procesos pendientes
  109 103
Procesador 1
Procesador 2
Procesador 3
  110 15 40
Procesador 4
  105 20 80
Procesador 5
  100 10 10
  102 1 1
  108 15 40
Procesador 6
Procesador 7
Procesador 8
  107 30 20
Procesador 9
  104 12 100
  106 7 10
Procesador 10
  101 41 20
Procesador 1
Procesador 2
Procesador 3
  110 15 29
Procesador 4
  105 20 69
Procesador 5
  108 15 29
Procesador 6
Procesador 7
Procesador 8
  107 30 9
Procesador 9
  104 12 89
Procesador 10
  101 41 9
Procesos pendientes
  109 103
Procesador 6
Procesador 6
  112 10 10
Procesador 4
  105 20 69

```

Procesador 4  
105 20 69  
Procesador 1

Procesador 2  
103 60 8  
109 30 20

## **Informació del problema**

Autor : PR02

Generació : 2018-12-12 15:44:50

© *Jutge.org*, 2006–2018.

<https://jutge.org>