

**SISTEMI OPERATIVI IDT/IEL/IIN**  
**INFORMATICA INDUSTRIALE E SISTEMI OPERATIVI IDI**  
**SISTEMI DI ELABORAZIONE p.o.**  
**prova scritta del 16 gennaio 2008**

Nome \_\_\_\_\_  
Cognome \_\_\_\_\_  
Matricola \_\_\_\_\_  
Corso di laurea \_\_\_\_\_

In un impianto per la produzione di motori sono presenti 3 stazioni di lavoro: una per la fusione del monoblocco (A), una per la tornitura dei pistoni (B), ed una per l'assemblaggio dei motori (C).

La logica che governa l'impianto prevede i seguenti vincoli:

- A non deve procedere con la fusione di monoblocchi se non esiste una scorta di pistoni pari ad almeno 4 unità;
- B non deve procedere con la tornitura di pistoni se la scorta di pistoni eccede le 12 unità;
- C assembla un motore impiegando un monoblocco e 4 pistoni.

Si delinei la soluzione per i tre processi che rappresentano A, B e C, usando i semafori Java come meccanismo di sincronizzazione, e cercando di non dar luogo a situazioni di attesa attiva.

Si tenga presente che la classe `Semaphore` di Java prevede, fra gli altri, anche i seguenti metodi:

- `public void acquire(int permits) throws InterruptedException`  
con questo metodo il thread invocante richiede il numero di gettoni (o permessi) specificato, eventualmente bloccandosi finché non sono tutti disponibili;
- `public void release(int permits)`  
con questo metodo il thread invocante restituisce al semaforo il numero di gettoni (o permessi) specificato.