

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 PROVA SCRITTA DI SISTEMI OPERATIVI
 ANNO ACCADEMICO 2012/2013
 12 settembre 2013

Esercizio -1: Essere iscritti su AlmaEsami per svolgere questa prova.

Esercizio 0: Scrivere correttamente nome, cognome, matricola e posizione in tutti i fogli prima di svolgere ogni altro esercizio. Scrivere esclusivamente a penna senza abrasioni. E' vietato l'uso delle penne cancellabili, della matita, dei coprenti bianchi per la correzione (bianchetto) e la scrittura in colore rosso (riservato alla correzione).
 Il compito e' formato da tre fogli, sei facciate compresa questa. Le soluzioni che si vogliono sottoporre per la correzione devono essere scritte negli spazi bianchi di questi fogli. Non verranno corretti altri supporti.
 E' obbligatorio consegnare il compito, e' possibile chiedere che esso non venga valutato scrivendo "NON VALUTARE" in modo ben visibile nella prima facciata.
 Per svolgere questo compito occorre solo una penna e un documento di identità valido. La consultazione o anche solo la disponibilità di altro materiale comporterà l'annullamento del compito (verrà automaticamente valutato gravemente insufficiente).

Esercizio c.1: Nel nostro problema ci sono una lavatrice e tanti utenti che desiderano lavare 1kg di biancheria a testa. La lavatrice puo' lavare da 3 a 5 kg di biancheria.

Quando la lavatrice e' libera occorrono almeno 3 kg di biancheria per iniziare il ciclo di lavaggio quindi occorre la biancheria di tre utenti. Quando il ciclo di lavaggio e' iniziato occorre portarlo a termine (non si puo' aggiungere biancheria).

Finito il lavaggio gli utenti ritirano la propria biancheria. Quando l'ultimo utente ha preso la propria biancheria, se vi sono almeno tre utenti in attesa si inizia un nuovo ciclo di lavaggio altrimenti occorre attendere l'arrivo di almeno tre utenti. Se ci sono tre o quattro utenti si fara' un ciclo di lavaggio di 3 o 4 kg, se ve ne sono almeno cinque si fara' un ciclo per 5kg. Se vi sono piu' di 5 utenti i rimanenti attenderanno il prossimo ciclo.

Scrivere un monitor lavatrice con due metodi enter e exit che regolino secondo le indicazioni appena date. La vita dei processi-utente puo' essere rappresentata dal seguente codice:

```
process utente[i] {
    while true {
        /* sporca biancheria */
        lavatrice.enter()
        /* lava biancheria */
        lavatrice.exit()
    }
}
```

Esercizio c.2:

Shared x=1 y=1 z=1	Process P { while(1) { x = x+1; } }	Process Q { while(1) { x = x+1; } }	Process R { while(1) { y = y+x; } }	Process R { while(1) { if (z==y) { print(z) z=0; } else z=z+1; } }
-----------------------------	---	---	---	--

Questo programma composto da 4 processi concorrenti e' soggetto a race condition.

Utilizzando alcuni semafori e inserendo nel codice solamente operazioni P e V su tali semafori e' possibile eliminare le race condition e fare in modo che l'unico output possibile sia la sequenza di tutti i quadrati dei numeri naturali: 1, 4, 9, 16, 25 etc.

Mostrare il programma modificato per ottenere questo risultato. (si ricorda che il quadrato di n e' anche la somma dei primi n numeri dispari).

Esercizio g.1: Si consideri il seguente scenario per un banchiere multivaluta.

(a) E' uno stato safe? Spiegate le motivazioni della vostra risposta.

(b) In caso positivo, mostrate una richiesta di allocazione che porterebbe il sistema in uno stato unsafe. In caso negativo, illustrate una richiesta di deallocazione che porterebbe il sistema in uno stato safe.

Valuta 1 COH = 5 Ci Pi Ni Max Corr. Residuo p1: 15 5 10 p2: 12 6 6 p3: 10 6 4 p4: 7 2 2 p5: 3 1 1	Valuta 2 COH = 25 Ci Pi Ni Max Corr. Residuo 30 5 25 100 65 35 50 15 35 20 10 10 70 35 35	Ci
---	---	----

Esercizio g.2: Le seguenti affermazioni sono vere o false? (Motivare dettagliatamente le risposte).

1. Il meccanismo RAID livello 0 aumenta la tolleranza ai guasti.
2. I microkernel consentono maggiori performance rispetto ai kernel monolitici
3. L'uso di librerie dinamiche consente una minor occupazione di memoria principale e secondaria.
4. Esistono stringhe di riferimenti per le quali LRU e MIN causano lo stesso numero di page fault.
5. Sia C la capability che consente l'accesso alla risorsa R. Per consentire che C sia memorizzata dal processo che usa R e' necessario che C sia crittata mediante un algoritmo a doppia chiave (Pubblica-Privata).

