

nome e cognome \_\_\_\_\_ numero di matricola 16 74 \_\_\_\_\_

CORSO DI SISTEMI OPERATIVI  
CORSO DI LAUREA IN INFORMATICA - UNIVERSITA' DI BOLOGNA  
SESSIONE AUTUNNALE 1998/99 - SECONDO APPELLO  
15 OTTOBRE 1999

**Esercizio -1.** Essersi correttamente iscritti per sostenere questa prova scritta (secondo le regole indicate in un apposito messaggio news).

**Esercizio 0.** Scrivere correttamente il proprio nome, cognome e numero di matricola in tutti i fogli.

Per lo scopo di questo compito chiamiamo semafori personalizzati un paradigma di programmazione concorrente dove le operazioni P e V hanno la sintassi/semantica spiegata qui di seguito.

Se un processo p esegue :

$P(s,q)$  -> fa una P verso il processo q (q deve effettuare  $V(s,p)$  oppure  $V(s,*)$ )

$P(s,*)$  -> fa una P verso chiunque (qualche altro processo dovrà richiamare  $V(s,p)$  oppure  $V(s,*)$ )

$V(s,q)$  -> fa una V verso q

$V(s,*)$  -> fa una V verso chiunque.

Lo stato iniziale è tale da rendere ogni P bloccante (concettualmente simile all'inizializzazione a zero dei semafori ordinari)

Le P e le V devono funzionare in modo coerente con la semantica tipica dei semafori: se a partire dallo stato iniziale p richiama  $V(s,q)$  e q richiama subito dopo  $P(s,p)$  oppure  $P(s,*)$  il processo q non si deve bloccare.

Le P e le V "mirate" devono essere prese in considerazione prima delle chiamate con : se un processo p richiama  $V(s,*)$  e ci sono due processi q in attesa per una  $P(s,*)$  e t in attesa per  $P(s,p)$ , allora t deve essere sbloccato (anche se è in attesa da meno tempo). E' vero anche il caso contrario: nella gestione di una  $P(s,*)$  se si può proseguire come effetto sia di una chiamata  $V(s,*)$  da parte del processo q e di una chiamata  $V(s,p)$  da parte di t, verrà "consumata" prima l'autorizzazione a proseguire data dal processo t.

La gestione delle richieste confrontabili (tutte quelle rivolte verso lo stesso processo o tutte quelle di tipo \*) viene effettuata in modo FIFO.

Esercizio 1. Implementare l'astrazione di semafori personalizzati tramite un monitor.



nome e cognome \_\_\_\_\_ numero di matricola 16 74 \_\_\_\_\_

Esercizio 2. Implementare un servizio di message passing sincrono facendo uso di un buffer condiviso fra tutti i processi e di semafori personalizzati (secondo la definizione dell'esercizio1). Si devono fornire le funzioni `send(p,m)` e `rv(p,m)`. Nella funzione di ricezione l'indicazione del processo mittente può essere sostituita dalla costante `*` quando viene atteso un messaggio da un mittente qualsiasi.

Esercizio 3. Si consideri l'istruzione atomica

$\text{boh}(x,y,z) ::= \langle x=y=y*z \rangle$

definita per valori di  $x$  reali e nell'intervallo  $[0,1]$ .

La funzione boh può essere utilizzata per realizzare un supporto per sezioni critiche (à la test&set)?  
perché?