

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA  
CORSO DI SISTEMI OPERATIVI  
ANNO ACCADEMICO 2009/2010 - PROVA DI CONCORRENZA  
ANNO ACCADEMICO 2010/2011 – Esercitazione Parziale di CONCORRENZA  
17 gennaio 2011

**Esercizio -1:** essersi iscritti correttamente 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 due fogli, quattro facciate compresa questa. Le soluzioni che si vuole 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. Questa prova puo' valere per la sessione straordinaria dell'AA2009/10 o come parziale dell'AA2010/11, indicare chiaramente quale opzione viene scelta nello spazio in alto a destra. (se non indicato, verra' valutato come errore dell'esercizio 0 e assegnato il valore di default 2009/10).

**Esercizio 1:** Scrivere un monitor reqq che gestisca una coda di richieste.

I richiedenti chiamano la funzione che ha la seguente signature:

```
answer_t reqq.query(request_t request);
```

Query deve fermare il processo richiedente fino a completamento della richiesta da parte di un gestore. Il valore di ritorno e' la risposta del gestore.

Ci sono N gestori che si comportano come segue:

```
multiq_handler: process[j, i=0,...,N-1] {  
    request_t req;  
    int type;  
    while (1) {  
        req=reqq.getquery(i);  
        reqq.reply(i,handle(req));  
    }  
}
```

Le richieste vengono assegnate ai gestori disponibili o accodate se sono tutti impegnati. Quando un gestore termina la gestione (funzione handle, che **non** implementare!) invia il risultato tramite la reply. Il valore passato alla reply deve essere restituito al richiedente come valore di ritorno della funzione query. Se non vi sono richieste disponibili i gestori si fermano attendendo nuove richieste.

**Esercizio 2:** Un semaforo frazionario ludico ha una sola chiamata:

```
class lfsem {  
    void frac(double x);  
private:  
    double value=1;  
}
```

Un semaforo frazionario ludico ha un valore privato (value) che deve essere sempre mantenuto maggiore o uguale a 1. Ogni chiamata a frac completata aggiorna il valore di value dividendolo per x, a patto che l'invariante venga mantenuto. Ogni chiamata a frac che non può venir completata sospende il processo chiamante. I processi bloccati vengono risvegliati in ordine fifo non appena il valore di value consenta il completamento dell'operazione.

Il semaforo frazionario ludico ha lo stesso potere dei semafori ordinari? (Si chiede dimostrazione!)

**Esercizio 3:** Sia dato un servizio di message passing asincrono.

Senza fare uso di processi server si implementi un servizio di message passing asincrono informato che fornisca le due seguenti funzioni:

```
int iasend(pid_t destination, T message)  
T iareceive(pid_t sender)
```

Il servizio si comporta come un ordinario servizio di msg passing con una sola eccezione: la iasend restituisce il numero di messaggi spediti verso *destination* che non risultano ancora ricevuti.

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA  
 CORSO DI SISTEMI OPERATIVI  
 ANNO ACCADEMICO 2009/2010 – PARTE GENERALE  
 17 gennaio 2011

**Esercizio -1:** essersi iscritti correttamente 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 due fogli, quattro facciate compresa questa. Le soluzioni che si vuole 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 (con la stessa penalizzazione di punteggio della grave insufficienza per la prossima esercitazione scritta).

**Esercizio 1:** Considerate un file system tipo UNIX inizialmente vuoto montato su /mnt.

Mostrate gli effetti delle seguenti operazioni facendo uno schema dello stato delle strutture dati nella situazione iniziale e come lo stato venga variato a seguito di ogni operazione. (si chiedono schemi spiegati, non temi!)

a- mkdir /mnt/dir1

b- ln -s /etc/passwd /mnt/passwd

c- echo ciao > /mnt/dir1/saluti

d- ln /mnt/dir1/saluti /mnt/arrivederci

e- rm /mnt/dir1/saluti

**Esercizio 2:** Sia data una stringa di riferimenti riferita a tre processi a,b,c. Si consideri un sistema che applichi il meccanismo del Working set in modo globale e non processo per processo. Il Working set ha durata equivalente a 15 riferimenti in memoria e il numero di frame globale del sistema e' pari a 10.

Sia data quindi la seguente stringa di riferimenti (a1 significa che il processo a accede alla pagina 1).

a1 b2 c3 a1 b4 c5 a6 b7 c5 a8 b9 c10 a1 b4 c5 a1 c5 b4 c3 b2 c11 (\*) a12 b7 a1 b4 b2 a1 b9

Calcolare i working set ad ogni istante. Cosa significa la situazione al punto (\*) e cosa fa/puo' aver fatto il sistema?

**Esercizio 3:** Sia x l'ultima e y la penultima cifra del vostro numero di matricola. Rispondete alla domanda  $(y*10+x)\%3$ .

0. Quali controlli effettua un file system check?

1. Come e' possibile revocare autorizzazioni fornite tramite capability?

2. Cosa e' e come funziona il processo init di un sistema UNIX?

NOTA: tutti gli esercizi verranno valutati solo se le risposte saranno corredate da motivazioni e dimostrazioni scritte in Italiano o in Inglese (corretto o almeno comprensibile). La presenza nella soluzione di un esercizio di solo codice sorgente/tabelle/scarabocchi/simboli vari comporta la non valutazione dell'esercizio.

