

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
CORSO DI SISTEMI OPERATIVI
ANNO ACCADEMICO 2010/2011 - PROVA DI CONCORRENZA

7 settembre 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 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.

Esercizio 1: Il monitor seq da implementare deve fornire due procedure entry:

procedure entry void enter(void);

procedure entry void exit(void);

I processi chiamano i servizi di seq come segue:

....

seq.enter();

/* codice controllato da seq */

seq.exit();

....

Tutti i processi che ne facciano richiesta possono entrare nel codice controllato da seq contemporaneamente (questa NON e' una sezione critica!). Exit deve garantire che i processi escano dal codice controllato rispettando la stessa sequenza di ingresso. Per esempio se tre processi a,b e c chiamano seq.enter in questo ordine b non potra' completare la exit se prima a non ha chiamato la exit. C dovra' attendere il completamento della exit sia di a sia di b per poter uscire.

Esercizio 2: Si considerino le seguenti funzioni atomiche foo e bar e si verifichi se e' possibile usarle (o meno) al posto della test&set per il supporto di sezioni critiche (i parametri sono passati per indirizzo):

2a) foo(x,y) = <x1=x+1, y1=y+1, x=y1, y=x1> (con x, y interi)

2b) bar(x,y) = <x = sqrt(y), y=abs(floatrand()+1> (con x,y reali, floatrand genera un numero random in tutto il dominio dei reali, abs e sqrt sono rispettivamente la funzione valore assoluto e la funzione radice quadrata).

Esercizio 3: Data una costante naturale N si implementi tramite semafori un servizio di una sola funzione:

void nbarrier(void)

che blocchi i primi N processi che chiamano la funzione quindi ne mantenga sempre N bloccati in modo che nessuno vada in starvation (a patto di avere un flusso costante di richieste a nbarrier da parte di piu' di N processi).

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 CORSO DI SISTEMI OPERATIVI
 ANNO ACCADEMICO 2010/2011 – PARTE GENERALE
 7 settembre 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: Elencare i controlli che deve effettuare un sistema di controllo di coerenza (fsck) per un file system di tipo unix (come minix o ext2). Per ogni controllo si indichi il metodo usato e le azioni che vengono effettuate in caso di errore.

Esercizio 2: Si considerino i grafi di Holt multirisorsa (ogni classe puo' contere piu' risorse). In questo caso (se la richiesta e' bloccante e le risorse sono non prerilasciabili e non condivisibili) la presenza di un Knot dovrebbe indicare lo stato di deadlock. Questa regola vale solo se ogni processo al al piu' una richiesta pendente o vale in ogni caso?

(Si richiede la dimostrazione o il controesempio. In ogni caso si richiede una chiara discussione del ragionamento che porta alla risposta).

Esercizio 3: Detti rispettivamente R il numero di riga e C il numero di colonna, soddisfare la richiesta corrispondente a $(R+C)\%3$ (esercizio obbligatorio):

- 0) si descrivano le peculiarita' di costruzione degli scheduler per sistemi multiprocessore
- 1) RAID e journaling: quali problemi risolvono? Sono da usare in alternativa o contemporaneamente? Perche'?
- 2) Algoritmi di rimpiazzamento e Working set: come interagiscono?

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.

