

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
 11 febbraio 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: Il monitor lifocs che lo studente deve implementare, realizza una critical section LIFO. I processi per usare la critical section LIFO usano il seguente protocollo:

```
n1=lifocs.enter()
... codice critico
n2=lifocs.exit()
```

quando un processo rilascia la sezione critica e ci sono processi in attesa di entrare, deve venir assegnata la sezione critica al processo che ha fatto l'ultima richiesta (in ordine di tempo).

Le funzioni enter e exit devono dare come valore di ritorno il numero di processi in attesa di entrare nella sezione critica.

I valori n1 e n2 sono correlati fra loro?

Esercizio 2: sia data la funzione atomica foo:

```
foo(void **x, void **y) {
    *x = *y;
    *y = x;
}
```

e' possibile usare foo al posto della test&set? (Se si': spiegare **bene** come funziona il codice presentato come soluzione, se no: spiegare **bene** i motivi dell'impossibilita').

Esercizio 3: siano dati seguenti quattro processi:

Process pin: while (1) { c=getchar(); fin(c); }	Process pout1: while (1) { fout_pre(1); print('A'); fout_post(1); }	Process pout2: while (1) { fout_pre(2); print('B'); fout_post(2); }	Process pout3: while (1) { fout_pre(3); print('C'); fout_post(3); }
---	--	--	--

I processi hanno memoria privata e possono fare uso di un meccanismo di message passing asincrono.

Scrivere le funzione fin, fout_pre e fout_post in modo che:

se il carattere letto da pin e' '1': venga posto in output "ABC",

se viene letto il carattere '2' l'output sia "BBB",

se viene letto il carattere '3' l'output sia "CAB" oppure "CBA" a seconda di quale processo fra pout1 e pout2 sia il piu' veloce. Per ogni altro carattere in input non vi e' alcun output.

Al termine dell'output la funzione fin esce e il processo pin puo' leggere il carattere successivo.

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 CORSO DI SISTEMI OPERATIVI
 ANNO ACCADEMICO 2009/2010 – PARTE GENERALE
 11 febbraio 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: Si consideri il seguente scenario per un banchiere multivaluta.

(a) Per quali valori di x lo stato e' safe? Motivare la risposta.

(b1) Scegliete un valore x per il quale lo stato sia safe e mostrate una allocazione che porterebbe il sistema in uno stato unsafe.

(b2) Scegliere un valore x per il quale lo stato sia unsafe e mostrare una deallocazione che porterebbe il sistema in uno stato safe.

Valuta 1				Valuta 2		
COH=5				COH=25		
i	Ci	Pi	Ni	Ci	Pi	Ni
	Max	Curr.	Residuo	Max	Curr.	Residuo
1	15	5	10	30	5	25
2	12	6	6	35+x	x	35
3	7	2	5	20	10	10
4	3	1	2	70	25	45
5	3	1	2	70	25	45

Esercizio 2: Sia data una memoria formata da 16KB inizialmente interamente non allocata.

Mostrare una (una sola!) sequenza di allocazione e deallocazione che contemporaneamente:

1. se gestita mediante paginazione con frame di 4KB provochi una perdita di spazio per frammentazione interna di 4KB (in totale)

2. se gestita mediante partizionamento dinamico con ogni politica di allocazione causi una perdita di spazio per frammentazione esterna sempre di 4KB

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

0- cosa e' e come funziona la memoria virtuale

1- cosa e' e come funziona un file system

2- cosa e' e come funziona uno scheduler della CPU

La risposta deve contenere almeno 100 parole (senseate).

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.

