

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
PROVA SCRITTA DI SISTEMI OPERATIVI
ANNO ACCADEMICO 2011/2012

18 settembre 2012

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 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 (verrà automaticamente valutato gravemente insufficiente).

Esercizio c.1: Per lo scopo di questo esercizio si ipotizzi la presenza di infiniti processi che si vogliono sincronizzare a tre a tre sulla base di un contrassegno numerico (tag).

Ogni processo chiama la procedura di ingresso syn3 del monitor three (three.syn3) passando come parametro (in input) un contrassegno numerico floating point (double) e un array di tre identificativi di processo (in output).

Monitor three:

Procedure entry void syn3(double tag, pid_t proc[3])

....

Solo quando tre processi hanno chiamato la syn3 specificando lo stesso valore per tag posso proseguire nella computazione. Il vettore proc conterra' gli identificativi dei tre processi (ogni processo quindi conoscerà l'identificativo degli altri due che hanno consentito di proseguire).

Se per esempio il processo P1 richiama syn3 con il valore 3.14, poi P2 con 1.41, P3 con 2.71, P4 con 3.14, se a questo punto arriva un processo P5 che chiama syn3 con valore 3.14 sblocca P1, P4 e P5 (P2 e P3 rimarranno in attesa che altri processi richiama syn3 con 1.41 o con 2.71). P1, P4 e P5 riceveranno nel vettore proc gli identificativi di P1, P2 e P4.

Esercizio c.2: Sia dato un servizio di message passing asincrono.

Scrivere un servizio di message passing asincrono a messaggi doppi (implementare quindi le chiamate dsend drecv): la drecv deve consegnare un messaggio solo quando due messaggi con lo stesso contenuto vengono ricevuti da due mittenti distinti.

Esercizio g.1: Sia dato un file system di tipo UNIX. Può essere soggetto a diversi tipi di incoerenza. Per ogni tipo indicare come riconoscere l'incoerenza e le azioni correttive operate da fsck.

- 1- errato numero di blocchi del file
- 2- errato numero di link di un file (reference count)
- 3- errori nella tabella/bitmap dei blocchi occupati/liberi
- 4- problemi di connettività del file system

Esercizio g.2: Rispondere ad almeno due delle seguenti domande:

- A cosa serve l'algoritmo di Dekker e come funziona?
- Access Control List/Capability, a cosa servono? Quali sono le differenze?
- Cosa significa "partizionare" un disco? A cosa serve?