

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 PROVA SCRITTA DI SISTEMI OPERATIVI
 ANNO ACCADEMICO 2022/2023
 13 giugno 2023

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 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.

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: Scrivere il monitor rb (redblack) con due procedure entry:

`float meanblack(float v)`

`float meanred(float v)`

Esistono due tipi di processo, neri e rossi. I processi neri chiamano la funzione `meanblack` mentre i processi rossi chiamano `meanred`.

Entrambe le funzioni restituiscono la media dei valori passati da una chiamata di `meanblack` e una di `meanred`.

Per fare il calcolo ogni chiamata di un processo rosso (`meanred`) deve sincronizzarsi con una chiamata di un processo nero (`meanblack`) e viceversa. Se la prima chiamata è una `meanblack` il processo chiamante attende, quando successivamente un processo rosso chiama `meanred` entrambi i processi si sbloccano ed entrambe le funzioni devono restituire lo stesso valore (la media dei valori del parametro `v`). Se non arrivano chiamate di `meanred` i processi neri che chiamano `meanblack` devono attendere in ordine FIFO. Lo stesso vale anche per i processi `meanred` fino ad una chiamata di `meanblack`.

Il monitor rb deve usare una sola variabile di condizione e nessuna coda/lista/array, solo valori scalari (int o float).

Esercizio c.2: Dato un servizio di message passing asincrono costruire un servizio di message passing completamente asincrono a liste che abbia due metodi:

`nbl_send(msg_t m, pid_t dest)`

`list of msg_t nbl_receive(pid_t sender)`

La chiamata `nbl_receive` deve restituire la lista di tutti i messaggi ricevuti dal mittente specificato (`sender`) o la lista di tutti i messaggi ricevuti se `sender` è -1 (ANY).

Il servizio è completamente asincrono perché se non vi sono messaggi che soddisfano la condizione richiesta `nbl_receive` restituisce una lista vuota senza attendere.

Esercizio g.1: In un file system di tipo FAT i primi due blocchi dati (indice 0 e 1) e i corrispondenti elementi della tabella FAT sono riservati. In un certo istante la tabella FAT di una partizione con 14 blocchi dati contiene:

Res, Res, EOC, EOC, 4, EOC, 5, EOC, 12, EOC, EOC, 10, 13, 7

(Res significa riservato, EOC è il tag di end of chain, la fine della lista)

Il primo blocco libero ha indice 8

Il blocco dati di indice 2 contiene la root directory: il file 'f' ha l'indice del primo blocco 3, il file 'g' inizia dal blocco 6 e la directory 'd' dal blocco 9.

Il blocco dati 9 contiene una directory: il file 'd1' inizia dal blocco 10, il file 'd2' dal blocco 11 e il file 'd3' dal blocco 7.

Il file system ha alcune incoerenze, trovare quali sono e indicare quali operazioni potrebbe compiere un programma fsck per ripristinare la coerenza.

Esercizio g.2: rispondere alle seguenti domande (*motivando opportunamente le risposte!*):

- La libreria C fornisce delle funzioni che consentono di richiedere system call (per esempio open, read, etc). Come sono realizzate queste funzioni? Come vengono passati i parametri e il valore di ritorno delle system call?
- Dati 4 dischi da 1GiB, si possono memorizzare più dati usando una organizzazione RAID10 o RAID5? Perché?
- Dimostrare che se un algoritmo di rimpiazzamento è a stack allora non può soffrire della anomalia di Belady
- L'algoritmo del Banchiere evita che si verifichino situazioni di deadlock. Sembra una funzione desiderabile eppure viene raramente usato nei sistemi reali, perché?