

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA  
CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2007/2008  
CONCORRENZA - 14 Gennaio 2009

**Esercizio -1:** essersi iscritti correttamente per svolgere questa prova.

**Esercizio 0:** Scrivere correttamente nome, cognome, matricola e posizione prima di svolgere ogni altro esercizio.

**Esercizio 1:** Scrivere un monitor che implementi un buffer limitato (SIZE elementi) dove i messaggi abbiano associata una priorit . I processi produttori richiamano

**pbb.enqueue(m,prio)**

mentre i processi consumatori chiamano

**m=pbb.dequeue()**

I primi SIZE processi produttori pongono i loro elementi nel buffer senza bloccarsi, i successivi si fermano in attesa.

Il consumatore deve sempre prendere l'elemento a priorit  massima fra quelli nel buffer \*e\* fra quelli in attesa.

**Esercizio 2 (Test&Set):** Quali delle seguenti funzioni possono essere utilizzate per implementare un meccanismo di sezione critica e in che modo?

divmult1(int x, int y) =<x=x\*y, y=x/x>

divmult2(int x, int y) =<x=x\*y, y=x/y>

divmult3(int x, int y) =<xx=x, x=x\*y, y=xx/x>

divmult4(int x, int y) =<xx=x, x=x\*y, y=xx/y>

**Esercizio 3:** Scrivere un servizio di message passing asincrono (quello classico che consenta alla arecv di aspettare un messaggio da un utente specifico o da tutti usando \*\*) sulla base di un servizio asincrono **asendx/arecvx**, dove la **arecvx** non ha parametri e riceve FIFO tutti i messaggi (come se fosse **arecv** (\*)).



UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA  
CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2007/2008  
PARTE GENERALE - 14 Gennaio 2009

**Esercizio -1:** essersi iscritti correttamente per svolgere questa prova.

**Esercizio 0:** Scrivere correttamente nome, cognome, matricola e posizione prima di svolgere ogni altro esercizio.

**Esercizio 1:** Sia dato questo file system FAT (root dir entry point 0)

FAT  
0 NULL  
1 NULL  
2 5  
3 NULL  
4 NULL  
5 4

Blocco dati:

0-Dir: (etc 1 1) (vmlinux 2 4)  
1-Dir: (passwd 3 3)  
2-ea66 0008 0000 07c0 c88c d88e c08e d08e ....  
3-root:x:0:0:root:/root:/bin/bash\ndaemon:x:1:1:daemo....  
4-e2d0 5cff 07fc 8726 cf3e 25a4 0023  
5-2eeb 6448 5372 0204 0000 0000 1000 1212 .....

Trasformare il file system in uno equivalente (con la stessa struttura gerarchica e con gli stessi contenuti) ma ext2.

**Esercizio 2:** Un sistema operativo utilizza un algoritmo second chance (o algoritmo dell'orologio) come algoritmo di rimpiazzamento. Rispondere alle due seguenti domande:

a) Mostrare cosa accade nel caso della seguente stringa di riferimenti. Nota: tutte le informazioni utilizzate dall'algoritmo (p.e. la posizione della lancetta) debbono essere evidenziate nella risposta.

1 2 3 4 5 3 1

b) Estendere la stringa di riferimento del punto a utilizzando solamente pagine nell'intervallo [1,5] in modo tale che, una volta processate tutte le richieste, nel frame 1 sia caricata la pagina 1, nel frame 2 la pagina 2, nel frame 3 la pagina 3 e nel frame 4 la pagina 4.

**Esercizio 3:** Sia x l'ultima e y la penultima cifra del vostro numero di matricola. Confrontate le due soluzioni elencate al punto  $(y*10+x)\%4$ . In particolare: descrivere il problema che risolvono (A); descrivere brevemente le due soluzioni (B); mettete in evidenza i vantaggi (C), gli svantaggi (D) e i limiti di utilizzo (E) di ciascuna di esse.

0. User Thread vs Kernel Thread
1. SJF vs Round Robin
2. Look vs Clook
3. Best Fit vs First Fit



