

Sistemi Operativi L-A
Simulazione di prova di laboratorio 12 giugno 2009

1. Esercizio di programmazione di sistema tramite system calls

Si scriva un programma in C che, utilizzando le *system call* di Unix, preveda la seguente sintassi:

esame N M C filein

dove:

- **N** e **M** sono interi positivi, tali che $M > N$;
- **filein** è il nome di un file esistente e leggibile;
- **C** è un singolo carattere.

Una volta invocato, il programma deve:

- attivare un processo 'padre', P0, che attivi N processi figli.
- ciascun figlio P_i ($i=1, \dots, N$) deve leggere il file **filein** per analizzare un blocco di i caratteri ogni M letti. Per ogni carattere K così campionato si verifica se K è diverso da **C**: in questo caso il figlio deve comunicare al padre il carattere K ; in ogni caso, allo scadere di **M** secondi dalla propria creazione, ciascun figlio deve terminare la propria esecuzione.
- il padre deve attendere la terminazione di tutti i figli, e quindi stampare a video il numero totale di caratteri ricevuti dai figli.

2. Esercizio di programmazione concorrente in Java

Si realizzi un sistema cambia monete per una sala da gioco.

Il sistema deve permettere a un numero arbitrario di clienti concorrenti, ciascuno rappresentante un utente, di cambiare monete/banconote in gettoni per il gioco (fiches) e viceversa .

Ogni gettone vale 1 € .

Il sistema accetta solo banconote da 10 €.

I cambi possono avvenire in due modi:

- un utente presenta 1 banconota da 10 € e ottiene 10 gettoni
- un utente presenta N gettoni (N multiplo di 10) e ottiene N/10 banconote (per un totale di N €).

Si sviluppi un'applicazione java che, rappresentando ogni cliente come un thread concorrente, e utilizzando i semafori come strumenti di sincronizzazione tra thread, realizzi il sistema.

3. Esercizio di programmazione Shell Unix

Si realizzi un file comandi Unix con la seguente interfaccia:

trovaFile <utente> <dir>

dove <dir> è un **direttorio assoluto** esistente nel filesystem e <utente> un qualunque nome utente.

Dopo aver effettuato gli opportuni controlli sugli argomenti passati in ingresso, il file comandi dovrà esplorare la gerarchia di sottodirettori del direttorio <dir> e stampare a video il **nome assoluto** di tutti i file regolari il cui utente proprietario sia quello indicato dalla stringa <utente>.