

Esame di **Sistemi Operativi** DM270 dell'8 settembre 2011
Durata della prova: 2 ore e 45 minuti

Cognome e Nome		Matricola	
----------------	--	-----------	--

TRACCIA A

Prova 1

Si supponga di disporre di un disco composto da 1000 cilindri numerati da 0 a 999. Il disco sta servendo una richiesta relativa al cilindro 300 e la richiesta precedente era relativa al cilindro 200, la coda di richieste inevase in ordine FIFO è composta delle seguenti richieste:

100, 500, 900, 150, 400, 250, 350, 50, 750

assumendo come punto di partenza la posizione attuale della testina, calcolare la distanza totale (in cilindri) che il braccio del disco percorre per soddisfare tutte le richieste inevase usando i seguenti algoritmi di scheduling:

- FCFS;
- C-SCAN.

Prova 2

Si descriva, anche con l'aiuto di frammenti di codice opportunamente commentati, come si effettua la creazione e l'esecuzione di processi utilizzando le system call di Unix.

Prova 3

Un casello autostradale è composto da N porte dove i veicoli pagano un pedaggio calcolato secondo una tariffa chilometrica T . Il casello include una variabile *incasso* che memorizza l'ammontare di denaro riscosso nella giornata attraverso tutte le porte.

Ogni veicolo effettua una volta sola le seguenti operazioni:

1. percorre un tratto di autostrada di x chilometri (con x compreso tra 50 e 100), impiegando 40 secondi per ogni chilometro;
2. giunto al casello, sceglie a caso la porta p a cui accodarsi;
3. accede alla porta p , dopo aver atteso in ordine FIFO che i precedenti veicoli abbiano completato le operazioni di pagamento su quella porta;
4. impiega tra 3 e 6 secondi per effettuare il pagamento;
5. rilascia la porta p ed abbandona l'autostrada.

Ad ogni pagamento, la variabile *incasso* del casello deve essere incrementata di una quantità pari a $x * T$.

Si modellino in Java i veicoli attraverso dei Thread e si implementino due soluzioni che riproducano il funzionamento del problema sopra descritto utilizzando:

1. la classe *Semaphore* del package *java.util.concurrent*
2. gli strumenti di mutua esclusione e sincronizzazione del package *java.util.concurrent.locks*.

Si scriva infine un *main* d'esempio che faccia uso di una delle due soluzioni precedenti. A tal fine si inizializzi un certo numero V di veicoli, assegnando ad N ed a T dei valori a scelta dello studente.

NB: Non si possono utilizzare appunti ed altro materiale durante lo svolgimento dell'esame.