

## Paradigmi Concorrenti e Distribuiti Sistemi operativi e reti (complementi)

Appello del 23 settembre 2009

Cognome \_\_\_\_\_ Nome \_\_\_\_\_ Mat. \_\_\_\_\_

QUESTO FOGLIO VA CONSEGNATO COMUNQUE, CON L'INDICAZIONE: **RITIRATO** ☐ **CONSEGNATO** ☐

Una base di dati viene consultata e aggiornata da  $N$  processi. Ogni processo accede ciclicamente alla base di dati specificando, per ogni operazione, se voglia accedere in lettura o in scrittura. Si vuole garantire che le operazioni di scrittura avvengano con accesso esclusivo; le operazioni di lettura possono invece essere simultanee fra loro (ma non con quella di scrittura). Nel caso in cui sia stata richiesta una scrittura, il sistema deve consentire che si concludano le letture in corso, ma deve poi dare la precedenza alla scrittura. Un processo di controllo, indipendente dal monitor della base di dati, può, in qualunque momento, ordinare a uno dei processi di cessare l'attività; se il processo in questione ha un'operazione in corso, la porta a termine prima di arrestarsi.

Si chiede di modellare il sistema descritto come composizione parallela di FSP, ponendo  $N=4$ , e di tradurre il modello in un programma concorrente, in Java.

**\*\*\* Le soluzioni vanno commentate adeguatamente. \*\*\***

### Regole per l'esame

- Obiettivo dell'esame è dimostrare la capacità di comprendere un semplice problema di concorrenza, di saper ideare una soluzione e di esprimerla attraverso gli strumenti del corso. In particolare, il modello a stati con LTSA deve trattare gli aspetti di interazione e concorrenza tra i processi, trascurando, dove non necessari, gli aspetti relativi alle attività dei processi. Il programma Java deve essere l'impostazione, il prototipo, di un possibile sistema realizzato. Anche in questo caso interessano più gli aspetti di interazione e concorrenza, anche se il sistema deve essere il più possibile completo.  
**È richiesto che sia la parte LTSA sia la parte Java siano sufficienti per superare l'esame.**  
**È richiesto che la realizzazione Java sia conforme al modello realizzato con LTSA.**
- Durante lo svolgimento dell'esame si possono usare le risorse elettroniche presenti sul computer e nel proprio spazio personale. NON si possono usare appunti, libri, manuali o altri strumenti esterni.
- Le soluzioni devono essere commentate in modo da poter comprendere ciò che non è evidente dal codice (per esempio, il significato delle etichette e delle variabili, le cause di sospensione e risveglio dei thread).
- La correzione del compito avviene in modo elettronico per la parte LTSA (quindi il sistema deve essere compilabile ed eseguibile, anche se, ovviamente, potrebbe essere non completo o parzialmente sbagliato), ed attraverso il solo esame del codice per la parte Java (quindi è importante che si capisca bene la struttura del programma, mentre sono secondarie istruzioni che servono solo alla visualizzazione del comportamento – es. sleep, yield, println, ...).
- **Contestualmente alla restituzione del compito viene discussa la soluzione proposta. Si suggerisce quindi di arrivare alla discussione consapevoli di quanto fatto e possibilmente con una versione corretta e completa del compito.**  
**All'atto della restituzione verrà fatta una breve discussione pubblica della soluzione ad uso di chi non ha superato l'esame e di quanti devono ancora sostenerlo.**
- Il risultato dell'esame può essere di tre tipi. (a) esame superato, quindi viene registrato il voto (NB: i voti vanno da 18 a 30elode, quindi anche soluzioni parziali possono portare alla sufficienza, se queste dimostrano la comprensione degli aspetti fondamentali); (b) esame non superato, lo studente deve ripetere la prova senza penalità; (c) esame gravemente insufficiente, lo studente deve ripetere la prova saltando un appello (NB: la sanzione ha lo scopo di scoraggiare lo studente che intende "provare a passare l'esame". Si consiglia quindi di consegnare il compito solo se si ritiene che possa essere sufficiente).

### Per la soluzione in LTSA:

- 1) Assegnare al proprio file .Its il nome COGNOMEMATRICOLO.Its (es. Rossi123456.Its N.B. mettete la vostra matricola !!! **non** chiamatelo in altri modi come 123456.Its o 123456Rossi.Its).
- 2) Nel file .Its aggiungete come prime due righe di commento matricola e nome/cognome  
(es. //Tizio DeCaio  
//matr.123456 )

### Per la soluzione in Java:

- 3) Inserire tutte le classi che formano la soluzione in **un unico file**, di nome COGNOMEMATRICOLO.java (es. Rossi123456.java). [NB: Quindi la classe che contiene il main deve chiamarsi, come il file, COGNOMEMATRICOLO]
- 4) Nel file .java aggiungete come prime due righe di commento matricola e nome/cognome  
(es. //Tizio DeCaio  
//matr.123456 )

### Per consegnare le soluzioni

- 5) Dalla Barra di Comando selezionate Run e nel campo Open digitate \\LIB\INFO\CORSI\SO\_COMPLEMENTI e dando OK vi si aprirà una finestra con la cartella CONSEGNA.
  - 6) Per consegnare il vostro file .Its dovete trascinare l'icona del file sull'icona CONSEGNA e fare un Copy. Ripetere l'operazione per il file .java.
- N.B. NON POTETE LEGGERE IL CONTENUTO DELLA CARTELLA CONSEGNA E POTETE COPIARVI UNA VOLTA SOLA IL FILE, perciò state ben attenti a consegnare la versione giusta del vostro elaborato.**