

NON SPEGNERE IL PC A FINE ESAME

Corso di Sistemi Operativi e Reti

Corso di Sistemi Operativi

Prova scritta di SETTEMBRE 2017

ISTRUZIONI

1. Rinomina la cartella chiamata "CognomeNomeMatricola" che hai trovato sul Desktop e in cui hai trovato questa traccia, sostituendo "Cognome" "Nome" e "Matricola" con i tuoi dati personali;
2. Carica tutto il materiale didattico che vorrai usare sul Desktop; puoi farlo solo nei primi 5 minuti della prova;
3. Svolgi il compito; lascia tutto il sorgente che hai prodotto nella cartella di cui al punto 1;
4. Quando hai finito lascia la postazione facendo logout,

senza spegnere il PC.

SALVA SPESSO il tuo lavoro

PER GLI STUDENTI DI SISTEMI OPERATIVI: si può sostenere solo uno dei due esercizi se si è già superata in un appello precedente la corrispondente prova. Il tempo a disposizione in questo caso è di 2 ORE.

NON SPEGNERE IL PC A FINE ESAME

ESERCIZIO 1 (Programmazione multithread. Punti: 0-20)

Bisogna progettare delle classi che raffinano le classi `Lock` e `Condition` di Java secondo le specifiche seguenti.

`LockEvoluto`:

la classe `LockEvoluto` si comporta come il Java `ReentrantLock` ma aggiunge i seguenti metodi:

`void lock(String key)`. **Acquisisce il lock in oggetto con chiave `key`.**

`void unlock(String key)`. **Viene rilasciato il lock, ma solo se `key` è identico alla chiave con cui il lock era stato precedentemente acquisito, altrimenti genera una eccezione `WrongUnlockKeyException`**

`Condition newStepCondition()`. **Crea e restituisce una nuova `StepCondition` appartenente al lock corrispondente (si veda sotto)**

La classe `StepCondition` deve implementare l'interfaccia `Condition` di Java, con l'aggiunta dei metodi:

`await(int volte)` . **Pone il thread chiamante in stato di wait. Da tale stato si esce dopo che il thread ha ricevuto un numero di signal pari a `volte`.**

`signal(int volte)` . **Deve essere equivalente all'invocazione di `signal` per un numero di volte pari a `volte`.**

NON SPEGNERE IL PC A FINE ESAME

CI SONO DEI PUNTI AMBIGUI NELLA TRACCIA? **COMPLETA TU**

È parte integrante di questo esercizio completare le specifiche date nei punti non esplicitamente definiti, introducendo o estendendo tutte le strutture dati laddove si ritenga necessario, e risolvendo eventuali ambiguità.

POSSO CAMBIARE IL PROTOTIPO DEI METODI RICHIESTI? **NO**

Non è consentito modificare il prototipo dei metodi se questo è stato fornito. Potete aggiungere qualsivoglia campo e metodo di servizio, e qualsivoglia classe ausiliaria, ma NON variare l'interfaccia dei metodi pubblici già specificati.

CHE LINGUAGGIO DEVO USARE? **JAVA 7 O SUCCESSIVO**

Il linguaggio da utilizzare per l'implementazione è Java. È consentito usare qualsiasi funzione di libreria di Java 7 o successivi.

MA IL MAIN() LO DEVO SCRIVERE? E I THREAD DI PROVA? **SOLO PER FARE IL TUO DEBUG**

Non è esplicitamente richiesto di scrivere un `main()` o di implementare esplicitamente del codice di prova, anche se lo si suggerisce per testare il proprio codice prima della consegna.

NON SPEGNERE IL PC A FINE ESAME

ESERCIZIO 2 (Linguaggi di scripting. Punti 0-10)

Il comando bash `cat /proc/meminfo` visualizza alcune informazioni riguardo la memoria

Esempio 1

Un possibile output del comando `cat /proc/meminfo` è:

```
MemTotal:      8167848 kB
MemFree:       1409696 kB
Buffers:       961452 kB
Cached:        2347236 kB
...
```

In questo esercizio bisogna realizzare uno script Perl, che verrà salvato nel file `memory_script.pl`. Lo script accetta un parametro opzionale su linea di comando (che chiameremo `TIME`), che rappresenta un tempo espresso in secondi, ed esegue il comando `C`, dove `C` = "`cat /proc/meminfo`", una volta ogni `TIME` secondi. Se il parametro `TIME` non è specificato, bisogna assumere un valore di default di **2** secondi.

Lo script dovrà girare finché non verrà interrotto con `CTRL-C`.

Ad ogni esecuzione del comando `C`, lo script deve visualizzare su Standard Output la differenza tra memoria libera (`MemFree`) registrata a questa esecuzione rispetto a quella dell'esecuzione precedente. Bisogna visualizzare il tutto seguendo il formato specificato nella sezione **Formato Output - Standard Output**.

Si assuma che inizialmente la memoria libera sia **0**.

Inoltre, ogni **5** volte che viene eseguito `C` (per esempio ogni 10 secondi (**5*2**) se non c'è nessun parametro specificato), lo script dovrà stampare, in modalità *append*, su un file chiamato `mem_free.log`, un report di tutte le differenze trovate. Tale report deve seguire il formato specificato nella sezione **Formato Output - Log File**.

Formato Output - Standard Output (DA RISPETTARE RIGOROSAMENTE)

Sia `D` la differenza numerica tra il valore di `MemFree` restituito da `C` durante l'esecuzione corrente e quella precedente. Se `D` è negativo bisogna stampare il carattere '<'; se invece `D` è positivo bisogna stampare il carattere '>'; se `D` è nullo bisogna stampare il carattere '='. Bisogna poi stampare uno spazio vuoto seguito dal valore di `D` con l'unità di misura (come riportato nell'output del comando `C`), seguito da un carattere di nuova linea '\n'.

Formato Output - Log File (DA RISPETTARE RIGOROSAMENTE)

Bisogna raggruppare gli eventi accaduti nelle ultime 5 esecuzioni di `C` per "tipo". I due tipi possibili sono gli "incrementi" e i "decrementi".

NON SPEGNERE IL PC A FINE ESAME

Si devono prima visualizzare i dati relativi agli incrementi, stampando il carattere 'i', seguito da un carattere di nuova linea '\n', seguito dalla lista di tutti gli eventi di incremento (separati dal carattere ','). Sul rigo successivo bisogna stampare il carattere 'd', seguito da un carattere di nuova linea '\n', seguita dalla lista di tutti i decrementi (separati dal carattere ','). Infine bisogna stampare sul rigo successivo 10 volte il carattere '-', seguito da un carattere di nuova linea '\n'.

Nota in Perl è possibile sospendere l'esecuzione per X secondi usando la primitiva 'sleep X'.

Nota Si può assumere che l'unità di misura sia sempre la stessa

Esempio Output - Standard Output

```
> 13384 kB
> 9844 kB
= 0 kB
< -2292 kB
< -372 kB
```

Esempio Output - Log File

```
i
13384,9844
d
-2292,-372
-----
```

NON SPEGNERE IL PC A FINE ESAME