

Prova Pratica 059 - online

La prova consiste di due esercizi, il primo è un esercizio di programmazione concorrente, il secondo è un esercizio bash.

Avrete a disposizione circa 50 minuti per risolvere i due esercizi e consegnarle la soluzione richiesta.

Potete usare il materiale contenuto in <http://www.cs.unibo.it/~ghini/didattica/TREE4OS2021.tgz> che dovete già avere scaricato prima di questo esame

Non potete navigare in internet cercando informazioni all'esterno.
Ovviamente, non potete comunicare con nessuno, in nessun modo.

Salvate i files spesso, per premunirsi in caso di crash.

Esercizio Esame Pratica Online 163 - rw2 (1/2)

Consideriamo il caso dei lettori-scrittori in cui tutti i thread accedono ad **una** stessa variabile intera, i lettori stampano a video la variabile, gli scrittori la incrementano di 1.

Tutti i thread eseguono un loop infinito, durante il quale i Lettori accedono alla variabile comune leggendone il contenuto e impiegando 2 secondi. Finita la lettura i lettori fanno una sleep per un tempo compreso tra 2 e 12 secondi e poi provano a leggere ancora. Gli Scrittori accedono alla variabile comune modificandone il contenuto ed impiegando 3 secondi. Finita la scrittura gli scrittori fanno una sleep per un tempo compreso tra 0 e 4 secondi e poi provano a scrivere ancora.

Quando uno Scrittore accede al buffer, nessun altro thread puo' accedere al buffer.

Rispetto all'esempio studiato a lezione, introduciamo un ulteriore vincolo: **ci sono due tipi di lettori, A e B, e i due tipi di lettori non possono leggere contemporaneamente**. In sostanza, **quando un lettore A legge, possono leggere assieme a lui solo altri lettori A ma non i B. Quando un lettore B legge, possono leggere assieme a lui solo altri lettori B ma non gli A**.

Quindi, il primo lettore che comincia a leggere determina quali altri lettori possono leggere assieme a lui. Se un lettore di un certo tipo (ad esempio A) cerca di leggere ma non può farlo perché in quel momento sta leggendo un lettore dell'altro tipo (B) allora il lettore di tipo A aspetta che la variabile sia libera da letture di tipo B e riprova ad accedere.

Per equità, quando uno scrittore termina la scrittura, dovrebbe consentire l'accesso sia ai lettori di tipo A che a quelli di tipo B.

Ci sono **3 lettori** di tipo A, **3 lettori** di tipo B e **2 scrittori**.

Implementare il sistema descritto, utilizzando dei thread POSIX per ciascuna figura (**lettore** e **scrittore**). Utilizzare come base dell'esercizio i seguenti files, aggiungendo il codice necessario:

`rw2.c` `DBGpthread.c` `DBGpthread.c` `prnterror.h`

CONTINUA IN PAGINA SEGUENTE

Esercizio Esame Pratica Online 163 - rw2 (2/2)

CONTINUAZIONE

Scrivere il Makefile per compilare e linkare i sorgenti. Inserire il necessario controllo di errore. In caso di errore grave, terminare il programma producendo un avviso a video.

Non è obbligatorio utilizzare, nel codice che aggiungerete voi, le funzioni DBG* contenute nei files DBGpthread.* ma è fortemente consigliato.

L'implementazione parte dal caso lettori e scrittori visto a lezione. Le differenze introdotte per gestire i due tipi di lettori sono queste:

I lettori sono di tipo 'A' oppure 'B' ma sono implementati da una stessa funzione Lettore.

La variabile globale intera **char tipoLettoreInLettura** indica quale è il tipo di lettori che stanno leggendo in quel momento. Questo valore può essere 'A' o 'B' ma deve essere considerato solo quando numLettoriInLettura>0 altrimenti non ha senso.

La var globale **pthread_cond_t condAttesaLettoriAltroTipo** serve per far attendere i lettori del tipo che non è correntemente in lettura.

Per implementare le attese di un tempo variabile, usare la funzione **attendi(int min, int max)**. La funzione **attendi(int min, int max)** implementata nel file **rw2.c** genera un numero casuale intero X compreso tra min e max ed attende X secondi prima di restituire il controllo al chiamante.

Le variabili globali già inserite sono già inizializzate.

Potete aggiungere variabili globali, se vi servono.

Ricordate di inizializzare le vostre variabili globali.

Esercizio Esame Pratica - 164 - estrai.sh

Scrivere un file di testo **input1.txt** in cui in ciascuna riga c'è una parola, seguita da una virgola, seguita da un numero intero maggiore o uguale a zero, seguito da una virgola, seguito da un'altra parola.

Ad esempio, il file **input1.txt** passato assieme al testo dell'esercizio contiene:

```
gatto,52,vacca  
alfa,31,triangolo  
orto,9,bici  
zappa,69,guzzi  
disco,12,celentano
```

Realizzare uno script **estrai.sh** che stampa a video le righe lette dal file **input1.txt** ma senza i numeri in mezzo a ciascuna riga e senza la virgola che segue il numero.

Infine, stampa a video la somma dei numeri che non ha visualizzato.

Qui sotto ecco cosa deve apparire a video usando il file **input1.txt** di esempio.

```
gatto, vacca  
alfa, triangolo  
orto, bici  
zappa, guzzi  
disco, celentano
```