

Esercizio buffer sincrono concorrente

Si richiede di implementare in C + POSIX Pthreads un buffer sincrono concorrente. Tale struttura dati concorrente potrà essere utilizzata per effettuare una comunicazione sincrona tra un insieme di thread produttori ed un insieme di thread consumatori.

Un buffer sincrono è una struttura dati in cui il thread che inserisce un item nel buffer (di tipo generico void*) aspetta fino a quando un thread consumatore abbia effettivamente prelevato l'item dal buffer. Analogamente, un thread consumatore che cerca di prelevare da un buffer sincrono un item, si blocca fino a quando l'item non viene prodotto da uno dei thread produttori.

Lo studente dovrà implementare la struttura dati buffer sincrono ed i metodi *put*, *get*, *tryput* e *tryget*. I metodi ritornano 0 se l'operazione è andata a buon fine, oppure -1 se si è verificato un errore, in quest'ultimo caso la variabile **errno** dovrà essere settata opportunamente. Le operazioni *tryput/tryget* vanno a buon fine se il buffer è vuoto/pieno, rispettivamente. Se tali operazioni falliscono per buffer pieno/vuoto, **errno** dovrà essere settato ad EAGAIN.

Il codice da consegnare deve essere contenuto in un tar file ed organizzato nel modo seguente:

- la struttura buffer sincrono ed i prototipi di interfaccia sono contenuti in un file .h (ad esempio syncbuffer.h);
- la definizione dei metodi deve essere contenuta in un file .c (ad esempio syncbuffer.c);
- dovrà essere fornito un programma *main* (ad esempio con nome syncbuffer-test.c), che linka la struttura dati buffer sincrono, ed in cui viene dimostrato il corretto utilizzo della struttura dati attraverso i metodi sopra elencati con un numero di produttori e consumatori specificato a linea di comando;
- deve essere definito un Makefile con target test che permette di compilare il codice e lanciare i tests.