

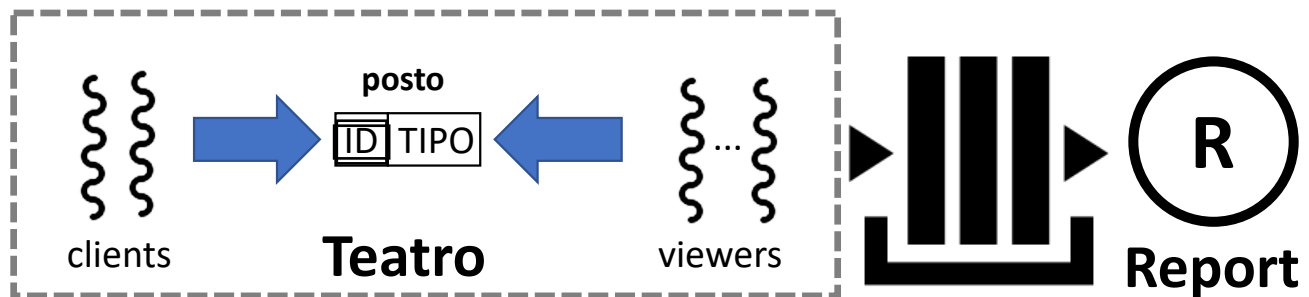
Testo della prova

Si realizzi in linguaggio C/C++ la seguente applicazione **multithread** (PThread) basata sul costrutto **monitor** e **code di messaggi**. La applicazione dovrà simulare l'aggiornamento in tempo reale delle prenotazioni di un teatro per i posti in platea e in balconata.

Un processo **Teatro** gestisce la prenotazione dei posti (identificati da ID e Tipo). Tale processo è costituito da **2 thread client** che scrivono (ogni secondo) il numero del posto da occupare (un intero casuale tra 0 e 100) e la tipologia (un intero pari a 0=platea o 1=balconata) su un buffer condiviso con altri **5 thread viewer** che possono effettuare la lettura (ogni secondo) dei posti e della tipologia scritti. La sincronizzazione sul buffer condiviso deve essere gestita tramite schema **lettori-scrittori con starvation di entrambi**.

Una volta che i **thread viewer** effettuano la lettura del posto, dovranno inviare, tramite coda di messaggi asincrona, il tipo dei posti analizzati ad un processo **Report**. Tale processo attenderà la ricezione di 20 messaggi e stamperà a video il numero di posti in platea e in balconata che sono stati ricevuti.

Si creino **2 eseguibili distinti**, uno per il processo **Teatro** e uno per il processo **Report** da lanciare attraverso un ulteriore applicazione (master) tramite primitive della famiglia *exec*. I *thread client* del processo **Teatro** dovranno scrivere 10 prenotazioni di posto e tipologia random, per un totale di 20 scritture. Ogni *thread viewer* leggerà 4 prenotazioni di posto, per un totale di 20 letture. Il processo **Report** appena riceverà 20 messaggi di prenotazioni di posto visualizzerà a video un conteggio dei posti platea e balconata che attualmente i *thread client* stanno cercando di prenotare.



File da completare:

- main.c
- teatro.h
- teatro.c
- teatro_procedure.c
- report.c