

[illegible]

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 della quotazioni di oro e argento.

```
typedef struct {  
    //aggiungere variabili per  
    la sincronizzazione  
    int quotazione_oro;  
    int quotazione_argento;  
}LettsCrit;
```

Un processo **Seller** gestisce le quotazioni in tempo reale. In particolare, tale processo è costituito da **2 thread seller worker** che scrivono la quotazione dell'oro e dell'argento su due buffer condivisi distinti. Un processo **Buyer** è costituito da **5 thread buyer worker** che possono effettuare la lettura (ogni 2 secondi) delle quotazioni dai due buffer condivisi. La sincronizzazione sui due buffer condivisi deve essere gestita tramite schema **lettori-scrittori con starvation di entrambi**, dove i thread seller fungono da scrittori e i thread worker da lettori.

Un volta che i thread **buyer worker** effettuano la lettura della quotazione, dovranno inviare, tramite coda di messaggi (una per l'oro e una per l'argento), le quotazione al processo **Report**. Tale processo accumulerà 10 messaggi di quotazioni e stamperà a video la media delle quotazioni.

Si creino **2 eseguibili distinti**, uno per i processi Seller e Buyer, e uno per il processo Report. I **seller worker** dovranno scrivere 5 quotazioni per l'oro e 5 per l'argento. I **buyer worker** leggeranno 2 volte le quotazioni per l'oro e 2 volte le quotazioni per l'argento. Il processo **Report** riceverà 10 messaggi di quotazione oro e 10 messaggi di quotazione argento.

Si crei un ulteriore applicazione (master) che lanci i 2 eseguibili descritti in precedenza.

