

*Università di Pisa -- Dipartimento di  
Informatica Corso di Laurea in Informatica*

**Progetto di Laboratorio di Sistemi Operativi**

a.a. 2019-20

Docenti: G. Prencipe (Corso A), M. Torquati (Corso B)

**Studente** : Iommi Andrea

**Matricola** : 578212

**Corso** : A

Note per l'esecuzione:

**make** /\* per compilare il progetto \*/

**make test2** /\* per far partire l'esecuzione \*/

**make result** /\* per visualizzare i risultati \*/

**make clean** /\* per cancellare i file temporanei creati nella compilazione \*/

# SUPERMERCATO

## Cliente-Cassiere

Le scelte di implementazione per l'interazione tra cliente e cassiere consistono: in una coda FIFO con accesso in mutua esclusione da parte di un cliente e dalla cassa. Ogni cassiera dispone di una sua coda dove i clienti in modo random possono accodarsi. La cassiera è stata implementata con una variabile che stabilisca se è disponibile per essere utilizzata o meno, nel periodo della sua disponibilità è tenuto un timer che, nel file delle statistiche determinerà il tempo di servizio.

La cassiera nel suo ciclo di vita, nel momento in cui il supermercato è aperto ma non dispone di clienti da processare, si mette in attesa. Il cliente successivamente alla fase di acquisti, attiverà la cassa scelta, così che la cassiera si svegli e processi tutti i clienti in coda per poi rimettersi in attesa.

Il cliente continuerà fino alla chiusura del supermercato cercando di trovare una cassiera disponibile che lo “servi”, altrimenti uscirà senza essere passato per una coda.

La cassiera terrà per ogni cliente processato una cronologia comune a tutte le casse, implementata con una semplice lista, che verrà successivamente utilizzata per formare il file di statistica.

## Cliente-Direttore-uno

Come da specifica il cliente potrebbe entrare nel supermercato ma non acquistare prodotti, in questo caso, il thread non sceglie una coda qualsiasi tra le casse, ma deciderà di mettersi in comunicazione direttamente con il direttore-uno (lista implementata sempre in mutua esclusione).

Il direttore-uno nel momento in cui avrà la sua coda non vuota, si attiverà e “servirà” il cliente con zero prodotti, inserendolo sempre nella medesima lista di clienti serviti dal supermercato.

## Cassiere-Direttore-uno

La cassiera terrà per ogni cliente servito una cronologia, implementata con una semplice lista, che verrà successivamente utilizzata per formare il file di statistica. Per questo tipo di attività ho optato per una variabile che contenga per ogni cassa il numero di clienti in quel momento presenti in coda, quest'ultima (un array delimitato sempre dalla mutua esclusione) viene aggiornata allo scadere dell'intervallo di tempo (generato da un timestamp) da ogni cassa.

L'interazione effettiva avviene nel momento in cui, dopo l'aggiornamento della coda, il direttore si risveglierà, quindi si preoccuperà di controllare il numero di clienti solo nell'ultima cassa, (che nella mia implementazione si presume sia quella da tenere in considerazione per l'apertura/chiusura) e secondo valori soglia prenderà le sue considerazioni.

- Apertura

L'apertura di una cassa viene implementata semplicemente dando la possibilità al consumatore nel momento della scelta, di un'alternativa e mettendo lo stato della cassiera come “Disponibile”.

- Chiusura

Reciprocamente la cassiera diventa “Non disponibile” e al consumatore viene tolta la possibilità di sceglierla.

Es ---->  $0 < \text{cassiera scelta} < \text{il numero di cassiere disponibili} - 1$

### Cassiere-Direttore-due

Nella mia implementazione il direttore è diviso in tre thread, uno dei quali si occupa di generare clientela nel momento necessario.

Il ciclo di vita del secondo direttore si compone in due fasi: il thread è in wait (durante la quale aspetta in modo passivo di essere risvegliato, o per terminare l'esecuzione) oppure il thread è in attività : nel caso in cui siano usciti più clienti di quelli consentiti, il direttore-due si preoccuperà di generare altri thread "consumatori" in un numero uguale al valore di soglia.

La decisione di far entrare altri clienti viene presa tramite una valutazione di una variabile che rappresenta il numero di consumatori attualmente presenti nel supermercato (la variabile viene utilizzata con accesso in mutua esclusione).

Il cassiere dopo aver servito ogni il cliente, accede in mutua esclusione al numero clienti presenti nel supermercato e, nell'eventualità che soddisfi le condizioni sufficienti richiamerà "thread-due" per la gestione della clientela.

Oss. Il controllo viene fatto dai cassieri,così che si riducono notevolmente il numero di "Signal".

### Direttore 3 - I segnali

Come ultimo attore, ho scelto di dedicare un thread alla gestione dei segnali ed alla corretta chiusura del supermercato per le due situazioni definite dalla specifica.

Quest'ultimo, al momento della creazione viene messo subito in attesa passiva di ricevere un segnale esterno (SIGQUIT, SIGHUP), così da non occupare tempo macchina e, al suo risveglio:

- SIGQUIT

Il segnale imposta una variabile (volatile sigatomic\_t) ad accesso atomico come "supermercato chiuso".

I clienti possono trovarsi in due stati, nella ricerca della cassa oppure in coda ad una di esse, nel primo caso vengono fatti uscire immediatamente (semplicemente controllando ciclicamente che il supermercato sia ancora aperto prima di accodarsi), nell'altro caso, il thread aspetta che sia il proprio turno alla cassa e dopo aver ottenuto la mutex esce non "comprando" i suoi articoli.

Le casse stesse possono trovarsi in due stati: in wait (la cassa non è disponibile oppure non ha clienti) o in attività, in entrambi casi il direttore-tre manda un segnale ad ogni cassa inizializzata, quindi la cassiera controlla la variabile "supermercato chiuso" ed esce.

Oss. Il direttore-uno è l'ultimo a terminare.

- SIGHUP

Il segnale imposta una variabile (volatile sigatomic\_t) ad accesso atomico come "clienti-stop".

In questa situazione il supermercato non chiude subito, ma aspetta di processare tutti i clienti al suo interno. Ovviamente il direttore-tre si occuperà di far uscire il direttore-due (colui che gestiva le entrate dei clienti), analogamente a prima, segnalerà ogni cassiera dell' avvenuto segnale. I clienti che stavano cercando un cassa non vengono fatti uscire, ma continuano a cercarne un disponibile. Per le casse la situazione può essere descritta da due stati: la cassa non ha clienti, quindi termina il turno(esce), altrimenti la cassa riceve il segnale, ma la sua coda non è vuota, quindi rimarrà attiva fino a quando non avrà più consumatori da servire.

La terminazione è garantita dal fatto che non possono generarsi più clienti processare.