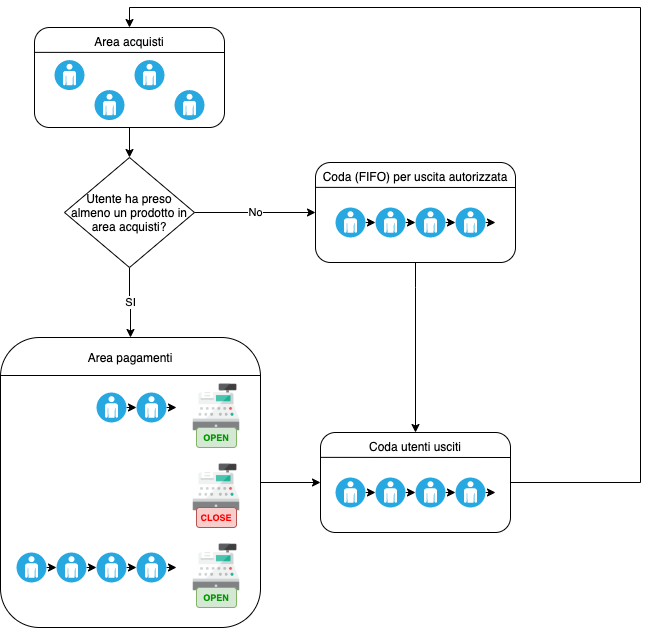
# Progetto di laboratorio

# Sistemi Operativi 2019/2020

Andrea Del Corto, matricola: 561446

# Scelte progettuali

La prima fase del progetto è stata quelle di definire chiaramente le entità coinvolte nel progetto. A tale scopo per prima cosa è stato realizzato il seguente schema che da una visione del supermercato incentrata in particolare sullo spostamento degli utenti all’interno del supermercato durante la simulazione:



Successivamente è stato definito l’insieme dei thread necessari per implementare il programma e i loro rispettivi compiti:

**Market thread**

Il programma ha un unico thread di questo tipo che gestisce i seguenti aspetti della simulazione:

1. Inizializza tutte le strutture dati necessarie per avviare la simulazione di un supermercato.
2. Quando in “Coda usciti” arrivano E utenti, si occupa di resettare i loro attributi per poi inserirli nuovamente in “Area acquisti”. Questo riutilizzo delle strutture dati permette di ridurre il numero di chiamate malloc e di creazioni di thread, con un conseguente riduzione del consumo delle risorse computazionali.
3. Quando un segnale di chiusura (SIGHUP o SIGQUIT) viene ricevuto, questo thread si occupa di gestire la chiusura facendo terminare in modo controllato tutti gli altri thread.

**User thread:**

Il programma ha C thread di questo tipo e ognuno di essi gestisce i seguenti aspetti della simulazione:

1. Gestisce una struttura dati “User” che contiene le informazioni relative ad un utente.
2. Registra timestamp di ingresso nell’area di shopping.
3. Attende un perido di tempo, definito in modo casuale compreso fra 10 e T millisecondi per simulare tempo di shopping.
4. La struttura dati “User” gestita dal thread viene spostata in una cassa aperta di “Area pagamenti” scelta casualmente se è stato preso almeno un prodotto, altrimenti viene spostata in “Coda (FIFO) per uscita autorizzata”.

**CashDesk service thread:**

Il programma ha K thread di questo tipo e ognuno di essi gestisce i seguenti aspetti della simulazione:

1. Gestisce una struttura dati “CashDesk” che contiene le informazioni relative ad una cassa.
2. Monitora i cambi di stato della cassa (aperta/chiusa) per tenere traccia del tempo totale di apertura e del numero di chiusure.
3. Ogni utente che viene inserito nella coda della cassa viene servito simulando un’attesa di servizio pari al seguente tempo:

|  |
| --- |
| T(n) = c1 + n\*NP    n: numero di prodotti presi dal ciente  c1: tempo costante diverso per ogni cassiere compreso fra 20-80 ms (viene determinato in modo casuale alla creazione della struttura dati CashDesk)  NP: è un parametro del file di configurazione che definisce il tempo in millisecondi necessario per processare un prodotto. |

**CashDesk notification thread:**

Ogni “CashDesk service thread” istanzia un thread ausiliario dedicato all’invio delle notifiche relative allo stato della cassa. Ogni TD millisecondi (TD è un parametro del file di configurazione), questo thread inserisce una apposita struttura dati contenente le informazioni di stato della cassa in una coda gestita dal thread direttore.

**Director desk handling thread:**

**Director authorization queue thread:**