

# PROGETTO DI LABORATORIO DI SISTEMI OPERATIVI A.A. 2019-20

NOME: ANTONIO      COGNOME: GUZZI  
MATRICOLA: 581947      CORSO: B

## FILE UTILIZZATI NEL PROGETTO

**DATASTRUCT.H:** È UN FILE HEADER IL QUALE CONTIENE LE STRUCT PRINCIPALI UTILIZZATE ALL'INTERNO DEL PROGRAMMA, TRA CUI UNA STRUCT CONTENENTE TUTTI I PARAMETRI PRESENTI NEL FILE "CONFIG.TXT".

**ANALISI.SH:** È LO SCRIPT TRAMITE IL QUALE VIENE ANALIZZATO IL FILE DI LOG NEL QUALE SCRIVE IL SUPERMERCATO AL TERMINE DELLA SUA ESECUZIONE. GRAZIE A QUESTO SCRIPT BASH SAREMO IN GRADO DI VISUALIZZARE A SCHERMO LE INFORMAZIONI UTILI PER QUANTO RIGUARDA LE CASSE E I CLIENTI.

**LIBSM.C/LIBSM.H:** È UNA LIBRERIA STATICA CONTENENTE LE FUNZIONI PRINCIPALI E PIÙ COMPLESSE UTILIZZATE DAL PROGRAMMA PER LA GESTIONE DELLE STRUTTURE DATI GLOBALI.

**LIBUTILS.C/LIBUTILS.H:** ANCHE QUESTA, COME LA PRECEDENTE, È UNA LIBRERIA STATICA, IN QUESTO CASO CONTERRÀ, PERÒ, FUNZIONI PIÙ SEMPLICI COME LA FUNZIONE DI APERTURA DEL FILE DI CONFIG E LE FUNZIONI UTILIZZATE PER LA LETTURA E LA MODIFICA DI VARIABILI GLOBALI.

**MAKEFILE:** È UN MAKEFILE TRAMITE IL QUALE VERRANNO AVVIATE ALCUNE FUNZIONI DI COMPILAZIONE ED ESECUZIONE DEL PROGRAMMA. TRAMITE QUESTO FILE SARÀ INOLTRE POSSIBILE RIMUOVERE TUTTI I FILE PRODOTTI ALLA FASE DELLA COMPILAZIONE.

**CONFIG.TXT:** È UN FILE .TXT A CUI VERRÀ ESEGUITO IL PARSING ALL'INIZIO DEL PROGRAMMA IN MODO TALE DA OTTENERE I PARAMETRI UTILI PER IL COMPIMENTO DI ALCUNE OPERAZIONI (COME AD ESEMPIO: TEMPO DI ELABORAZIONE DI UN PRODOTTO, VALORI DI SOGLIA, NUMERO DEI CLIENTI, NUMERO DELLE CASSE).

**LOGFILE.TXT:** È UN FILE DI LOG CHE VIENE SCRITTO/CREATO DAL SUPERMERCATO E CONTIENE TUTTI I PARAMETRI RACCOLTI DURANTE L'ESECUZIONE DEL PROGRAMMA. I PARAMETRI RIGUARDANO IL SUPERMERCATO, LE CASSE E I CLIENTI.

## IL SUPERMERCATO/IL MAIN

**IL SUPERMERCATO** È UNA FUNZIONE DEFINITA NEL FILE .C PRINCIPALE, QUEST'ULTIMA VIENE CHIAMATA ALL'INTERNO DEL MAIN E AL SUO INTERNO VENGONO ESEGUITE LE PTHREAD\_CREATE() DI TUTTI I THREAD NECESSARI PER L'ESECUZIONE DEL PROGRAMMA E LE RISPETTIVE PTHREAD\_JOIN().

**NEL MAIN** COME PRIMA COSA VIENE ESEGUITO UN CONTROLLO SUL NUMERO DI ARGOMENTI PASSATI AL PROGRAMMA, SE QUESTO RISULTA ESSERE DIVERSO DA 2, IL PROGRAMMA TERMINA CON UN MESSAGGIO DI ERRORE, ALTERNATIVAMENTE VIENE ESEGUITA LA FUNZIONE (CON GLI OPPORTUNI CONTROLLI) DEDICATA ALL'APERTURA E ALLA LETTURA DEL FILE DI CONFIG CONTENENTE TUTTI I DATI UTILI PER L'ESECUZIONE DEL PROGRAMMA. IN SEGUITO ALL'APERTURA DEL FILE, TRAMITE L'UTILIZZO DELLE FUNZIONI DI MEMSET() E SIGACTION(), IL MAIN SI OCCUPA DELL'INIZIALIZZAZIONE DI DUE STRUCT SIGACTION (UNA DEDICATA ALLA RICEZIONE DEL SEGNALE DI SIGQUIT E UNA ALLA RICEZIONE DEL SEGNALE DI SIGHUP) E ALLA INSTALLAZIONE DEI RISPETTIVI HANDLER (DI CUI PARLEREMO IN SEGUITO). PRIMA CHE VENGA EFFETTIVAMENTE ESEGUITA LA FUNZIONE CHE SI OCCUPA DI AVVIARE IL SUPERMERCATO, IL MAIN PROCEDE CON LA CREAZIONE DI DUE STRUTTURE DATI GLOBALI FONDAMENTALI PER L'ESECUZIONE DEL PROGRAMMA:

- UN ARRAY DI STRUCT TIPO CASHDESK\_INFO (STRUCT DEFINITA NEL FILE DATASTRUCT.H) LUNGO K, RAPPRESENTANTE LE CASSE PRESENTI ALL'INTERNO DEL SUPERMERCATO E I RISPETTIVI DATI (BIT DI APERTURA/CHIUSURA, PRODOTTI ELABORATI, CLIENTI SERVITI ... ). **N.B** NEL CASO DEL MIO PROGETTO HO DECISO CHE AL MOMENTO DELL'APERTURA DEL SUPERMERCATO TUTTE LE CASSE RISULTASSERO APERTE.
- UN ARRAY DI INTERI, ANCH'ESSO LUNGO K, INDICANTE LE LUNGHEZZE DELLE CODE DI CLIENTI PER OGNI CASSA. TALE STRUTTURA DATI VERRÀ UTILIZZATA DAL THREAD DEL DIRETTORE PER DETERMINARE LA CHIUSURA O L'APERTURA DI UNA CASSA.

SE LA CREAZIONE DELLE STRUTTURE DATI INTRODOTTE IN PRECEDENZA VA A BUON FINE, IL MAIN CHIAMA LA FUNZIONE DEL SUPERMERCATO E, ALLA SUA TERMINAZIONE, AVVIA UNA PROCEDURA PER LA SCRITTURA DELLE INFORMAZIONI RACCOLTE ALL'INTERNO DEL FILE DI LOG. PER LA SCRITTURA IN TALE FILE SARÀ NECESSARIA UNA LISTA AUSILIARIA GLOBALE, CONTENENTE TUTTI I DATI RELATIVI AI CLIENTI, CHE VERRÀ AGGIORNATA PRIMA CHE UN CLIENTE VENGA EFFETTIVAMENTE LIBERATO DURANTE LA SUA ELABORAZIONE, I DATI RACCOLTI DALLA LISTA VERRANNO POI SCRITTI ALL'INTERNO DEL FILE IN UN UNICO CICLO ALLA TERMINAZIONE DEI THREAD.

## GLI HANDLER DEI SEGNALI

LE FUNZIONI DEGLI HANDLER SI OCCUPANO PRINCIPALMENTE DI CAMBIARE LO STATO DEL SUPERMERCATO IN MODO TALE DA DETERMINARE LA SUA IMMINENTE CHIUSURA O CHIUSURA DEFINITIVA, COSÌ DA POTER ESEGUIRE OPERAZIONI DIFFERENTI PER ENTRAMBI GLI STATI.

- **HANDLER SIGQUIT:** DETERMINA LO STATO DI CHIUSURA DEFINITIVA DEL SUPERMERCATO, IL SUO STATO, RAPPRESENTATO DALLA VARIABILE GLOBALE `IS_CLOSED`, VIENE QUINDI SETTATO A 1.
- **HANDLER SIGHUP:** DETERMINA LO STATO DI CHIUSURA IMMINENTE DEL SUPERMERCATO, IL SUO STATO, ANCHE QUI RAPPRESENTATO DALLA VARIABILE `IS_CLOSED`, VIENE SETTATO A 2.

## IL CASSIERE

OGNI CASSA ALL'INTERNO DELLA PROPRIA STRUCT PRESENTA UNA MUTEX PERSONALE SULLA QUALE VIENE ESEGUITA LA LOCK PRIMA DI EFFETTUARE LE OPERAZIONI SULLA PROPRIA CODA, ANCH'ESSA CONTENUTA ALL'INTERNO DELLA STRUCT. QUESTO IN MODO TALE CHE IL PROGRAMMA POSSA EVENTUALMENTE LAVORARE SULLE ALTRE CASSE INDIPENDENTEMENTE DALLA LOCK PERSONALE OCCUPATA DALLA SINGOLA CASSA.

IL THREAD DELLA CASSA ESEGUE UN CICLO NEL QUALE LA GUARDIA CONTROLLA COSTANTEMENTE LO STATO DEL SUPERMERCATO (UTILIZZANDO LA VARIABILE GLOBALE “`IS_CLOSED`” E LA RISPETTIVA MUTEX “`MUTEX_IS_CLOSED`”), UN CAMBIAMENTO DI STATO PORTERÀ AD UN'USCITA DAL CICLO. ALL'INTERNO DEL CICLO LA CASSA CONTROLLA IL SUO STATO DI APERTURA O CHIUSURA:

- NEL CASO IN CUI LA CASSA RISULTI CHIUSA, ESEGUE UNA WAIT TEMPORANEA (PRECISAMENTE UNA `PTHREAD_COND_TIMEDWAIT()` DI 2 SECONDI) CHE SOSPENDE IL THREAD PER RIATTIVARLO DOPO UN QUANTO DI TEMPO LIMITATO E “FORZARLO” A CONTROLLARE NUOVAMENTE IL SUO STATO DI CHIUSURA O APERTURA.
- NEL CASO IN CUI LA CASSA RISULTI APERTA E LA SUA CODA DI CLIENTI È VUOTA, IL THREAD VIENE MESSO IN ATTESA TRAMITE L'UTILIZZO DI UNA VARIABILE CONDIZIONE FINO A QUANDO LA SUA CODA NON PRESENTA ALMENO UN CLIENTE AL SUO INTERNO.
- NEL CASO IN CUI LA CASSA RISULTI APERTA E LA SUA CODA DI CLIENTI NON È VUOTA, IL THREAD PROVVEDE ALL'ELABORAZIONE DI UN CLIENTE DELLA CODA TRAMITE L'UTILIZZO DELLA FUNZIONE `ELAB_CUSTOMER()` PRESENTE ALL'INTERNO DELLA LIBRERIA STATICA `LIBSM.C`.

**`ELAB_CUSTOMER()`** HA LA FUNZIONE DI EFFETTUARE UNA `NANOSLEEP` PER SIMULARE IL TEMPO DI ELABORAZIONE DEL CLIENTE, PER POI ESEGUIRE L'AGGIORNAMENTO DELLA LISTA CHE CONSERVA I DATI DEI CLIENTI E L'AGGIORNAMENTO DEI DATI DELLA CASSA (CONSERVATI NELLA SUA STRUCT) IN MODO TALE DA POTER ESSERE SCRITTI IN SEGUITO NEL FILE DI LOG. SUCCESSIVAMENTE AGLI AGGIORNAMENTI DELLE STRUTTURE DATI, `ELAB_CUSTOMER()`, ESEGUE UNA `POP()` SUL CLIENTE APPENA ELABORATO ED AGGIORNA LE VARIABILI INDICANTI: I CLIENTI TOTALI SERVITI (TRAMITE `INCR_SERVED_CUST()`), LA LUNGHEZZA DELLA CODA E I CLIENTI RIMOSSI DAL SUPERMERCATO (TRAMITE `INCR_REMOVED()`).

NEL MOMENTO IN CUI LO STATO DEL SUPERMERCATO CAMBIA, IL THREAD CASSIERE ESCE DAL CICLO, IN QUESTA FASE IL PROGRAMMA PUÒ TROVARSI IN DUE CASI:

- SE LO STATO DEL SUPERMERCATO È UGUALE A 1 IL THREAD CASSIERE SVEGLIA E LIBERA TUTTI I CLIENTI DAL SUPERMERCATO TRAMITE LA FUNZIONE `WAKE_UP_CUSTOMERS()` IMPEDENDO, COSÌ, L'ELABORAZIONE DEI CLIENTI RIMASTI DURANTE IL CAMBIO DI STATO DEL SUPERMERCATO.
- SE LO STATO DEL SUPERMERCATO È UGUALE A 2 IL THREAD CASSIERE SCORRE TUTTE LE CODE DELLE CASSE, INDIPENDENTEMENTE CHE QUESTE SIANO CHIUSE O APERTE, PER ESEGUIRE L'ELABORAZIONE DEI CLIENTI RIMASTI IN CODA. QUESTA OPERAZIONE VIENE FATTA INDIPENDENTEMENTE DALLLO STATO DELLA CASSA PER EVITARE CHE IL PROGRAMMA NON TERMINI A CAUSA DI CLIENTI RIMASTI IN CODA SU ALCUNE DELLE CASSE CHIUSE.

AL TERMINE DEL THREAD, TRAMITE UNA CHIAMATA DELLA FUNZIONE `CLOCK_GETTIME()`, VIENE FERMATO IL TEMPO DI APERTURA DELLA CASSA (AVVIATO ALL'INTERNO DELLA SUA FUNZIONE DI INIZIALIZZAZIONE) E SE QUESTA RISULTA ESSERE APERTA AL CAMBIO DI STATO DEL SUPERMERCATO, VIENE AGGIORNATA LA LISTA DEI TEMPI DI APERTURA CONTENUTA ALL'INTERNO DELLA STRUCT DELLA CASSA.

## IL SUPPORTO ALLE CASSE

ANCHE QUI, COME NEL THREAD CASSIERE, SI VERIFICA LO STATO DEL SUPERMERCATO ALL'INTERNO DELLA GUARDIA DI UN WHILE. FINCHÉ IL SUPERMERCATO RISULTA ESSERE APERTO, IL THREAD SUPPORTO DEL CASSIERE PROVVEDE AD AGGIORNARE PERIODICAMENTE L'ARRAY CONTENENTE LE LUNGHEZZE DELLE CODE DI CLIENTI DELLE CASSE. QUESTO THREAD, INOLTRE, COMUNICA CON IL DIRETTORE TRAMITE L'UTILIZZO DI UNA VARIABILE GLOBALE "INFO\_QUEUE\_MODIFY", QUEST'ULTIMA PERMETTE DI CAPIRE SE L'ARRAY DELLE LUNGHEZZE DELLE CODE È STATO MODIFICATO RECENTEMENTE OPPURE RISULTA ESSERE OBSOLETO, NEL MOMENTO IN CUI L'ARRAY RISULTA ESSERE AGGIORNATO RECENTEMENTE, IL THREAD DI SUPPORTO AL CASSIERE, VA IN ATTESA E ASPETTA DI POTERLO AGGIORNARE NUOVAMENTE. **N.B:** PER EVITARE DI RALLENTARE ECCESSIVAMENTE IL PROGRAMMA L'ARRAY DELLA LUNGHEZZA DELLE CODE E LA VARIABILE GLOBALE INDICANTE LA SUA MODIFICA, SONO GESTITI DALLA MEDESIMA MUTEX.

## IL CLIENTE

ALL'INTERNO DEL THREAD CHE GESTISCE IL CLIENTE VIENE INIZIALIZZATA LA STRUCT "CUSTOMER\_INFO" TRAMITE LA FUNZIONE INIT\_CUSTOMER(), NELLA QUALE VIENE ANCHE RANDOMIZZATO IL NUMERO DI PRODOTTI ACQUISTATI DAL CLIENTE.

SE IL NUMERO DI PRODOTTI ACQUISTATI DAL CLIENTE RISULTA ESSERE UGUALE A 0, ESSO VIENE INSERITO ALL'INTERNO DELLA LISTA GLOBALE "WAIT\_LIST" (GESTITA DALLA SUA MUTEX "MUTEX\_AUTHORIZATION") IN ATTESA DELL'AUTORIZZAZIONE DI USCITA DA PARTE DEL DIRETTORE, ALTERNATIVAMENTE IL CLIENTE ESEGUE UNA NANOSLEEP PER SIMULARE IL TEMPO DI SPESA E SCEGLIE IN MODO RANDOMICO UNA CASSA APERTA IN CUI INSERIRSI ACQUISENDONE LA RISPETTIVA LOCK, ESEGUENDO LA FUNZIONE DI INSERIMENTO NEL CASO IL SUPERMERCATO SIA APERTO, EFFETTUANDO INFINE UNA PTHREAD\_COND\_SIGNAL() SULLA CASSA, NEL CASO IN CUI QUEST'ULTIMA FOSSE ANDATA IN ATTESA ASPETTANDO CHE LA SUA CODA SI RIEMPISSA. DURANTE L'INSERIMENTO ALL'INTERNO DELLA CODA DELLA CASSA SCELTA IN PRECEDENZA, IL THREAD AVVIA IL TIMER PER MISURARNE IL TEMPO DI ATTESA IN CODA, IL QUALE VERRÀ FERMATO SOLO NELLA FUNZIONE DI ELABORAZIONE DEL CLIENTE "ELAB\_CUST()" ESEGUITA DALLA CASSA. UNA VOLTA INSERITO IL CLIENTE, IL THREAD CHE LO GESTISCE, RIMANE IN ATTESA FINO A QUANDO LA VARIABILE "MY\_TURN", INDICANTE LO STATO DEL CLIENTE, NON VIENE MODIFICATA DALLA FUNZIONE "ELAB\_CUST()" LA QUALE NE MUTA IL VALORE AD 1. IN SEGUITO AL CAMBIAMENTO DI STATO DEL CLIENTE VIENE RILASCIATA LA LOCK DELLA CASSA SCELTA PER L'INSERIMENTO E LA STRUTTURA RAPPRESENTANTE IL CLIENTE VIENE LIBERATA. **N.B:** LA VARIABILE DI TIPO CUSTOMER\_INFO\*, INIZIALIZZATA ALL'INIZIO DEL THREAD, POSSIODE IL MEDESIMO INDIRIZZO DELLA VARIABILE CHE VIENE PASSATA ALLA FUNZIONE DI INSERIMENTO NELLA CODA DELLA CASSA E CHE QUINDI È CONTENUTA IN ESSA, QUESTO PER FAR SÌ CHE LE MODIFICHE CHE VENGONO APPORTATE SULLA STRUCT DEL CLIENTE DA PARTE DELLA CASSA, SIANO VISIBILI ANCHE AL THREAD CHE GESTISCE IL CLIENTE STESSO. SENZA QUESTO TIPO DI OPERAZIONE, NON SAREBBE POSSIBILE NON SAREBBE POSSIBILE PER IL THREAD CLIENTE CONTROLLARE LO STATO DI "MY\_TURN".

## IL DIRETTORE

IL DIRETTORE UTILIZZA UN **THREAD SUPPORTO PER LA COMUNICAZIONE CON LA CASSA** CHE VERRÀ CREATO ALL'INTERNO DELLE SUE PRIME ISTRUZIONI. ANCHE IL DIRETTORE, COME I RESTATI THREAD PRINCIPALI DEL PROGRAMMA, UTILIZZA UN WHILE NEL QUALE VIENE CONTROLLATO LO STATO DEL SUPERMERCATO ALL'INTERNO DELLA SUA GUARDIA TRAMITE LA FUNZIONE "SUPERMARKET\_IS\_CLOSED()" CHE CONTROLLA LO STATO DELLA VARIABILE GLOBALE "IS\_CLOSED" UTILIZZANDO L'OPPORTUNA MUTEX. ALL'INTERNO DEL CICLO IL DIRETTORE CONTROLLA CHE I CLIENTI RIMOSSI DAL SUPERMERCATO SIANO MAGGIORI O UGUALI AL PARAMETRO "E" DEL FILE DI CONFIG, ANCHE QUESTO CONTROLLO AVVIENE TRAMITE UNA FUNZIONE SPECIALE (read\_removed()) IMPLEMENTATA ALL'INTERNO DELLA LIBRERIA "LIBUTILS.C": TALE FUNZIONE CONTROLLA "REMOVED", IL PARAMETRO GLOBALE CHE MONITORA IL NUMERO DI CLIENTI RIMOSSI DAL SUPERMERCATO, TRAMITE L'UTILIZZO DELLA MUTEX "MUTEX\_REMOVED". SE IL CONTROLLO RISULTA VERIFICATO, LA VARIABILE "REMOVED" VIENE AGGIORNATA IN MODO TALE CHE IL DIRETTORE AL CICLO SUCCESSIVO UTILIZZI DATI CONSISTENTI, IN SEGUITO VIENE CREATO UN ARRAY STATICO CONTENENTE I NUOVI TID DEI CLIENTI CHE VERRANNO AGGIUNTI AL SUPERMERCATO TRAMITE L'UTILIZZO DELLA PTHREAD\_CREATE(). I TID DEI NUOVI CLIENTI VENGONO INSERITI ALL'INTERNO DI UNA LISTA, QUESTO PER FARE IN MODO CHE, ESSENDO I NUOVI CLIENTI DEI SOTTO-THREAD DEL DIRETTORE, LA LORO JOIN VENGA ESEGUITA DIRETTAMENTE DAL THREAD DIRETTORE. PRIMA DEL TERMINE DEL CICLO IL DIRETTORE PERMETTE, INOLTRE, L'USCITA DEI CLIENTI CON 0 PRODOTTI IN ATTESA DELLA LORO AUTORIZZAZIONE, QUESTA OPERAZIONE È IMPLEMENTATA TRAMITE L'UTILIZZO DELLA FUNZIONE "EXIT\_CUSTOM()", LA QUALE

ESEGUE UNA `PTHREAD_COND_SIGNAL()` SUI THREAD CLIENTI CHE SI ERANO MESSI IN ATTESA DELL'AUTORIZZAZIONE DEL DIRETTORE.

UNA VOLTA USCITO DAL CICLO IL DIRETTORE ESEGUE LA SIGNAL DELLE CASSE E DEI CLIENTI IN ATTESA DELLA LORO AUTORIZZAZIONE IN MODO TALE CHE QUESTI VENGANO SVEGLIATI IN SEGUITO AL CAMBIAMENTO DI STATO DEL SUPERMERCATO, QUESTO PER EVITARE CHE I THREAD PRECEDENTEMENTE ELENCATI POSSANO TERMINARE.

## IL SUPPORTO AL DIRETTORE

ANCHE QUESTO THREAD, CREATO DAL DIRETTORE, LAVORA CON UN CICLO CHE PERMETTE DI CONTROLLARE LO STATO DEL SUPERMERCATO. ALL'INTERNO DEL CICLO SI ESEGUE UNA LOCK SULLA MUTEX CHE GESTISCE L'ARRAY DI LUNGHEZZE DELLE CODE DELLE CASSE ("`MUTEX_QUEUE_INFO`") E IL SUO BIT DI MODIFICA ("`INFO_QUEUE_MODIFY`"). SE L'ARRAY DELLE LUNGHEZZE DELLE CODE RISULTA ESSERE OBSOLETO, IL THREAD SUPPORTO AL DIRETTORE ESEGUE UNA WAIT E ASPETTA CHE QUEST'ULTIMO SI AGGIORNI, ALTERNATIVAMENTE VENGONO ESEGUITE LE OPERAZIONI DI CHIUSURA E APERTURA SULLE CASSE, IN BASE AI DATI PRESENTI ALL'INTERNO DELL'ARRAY "`INFO_QUEUE`":

- **FUNZIONE DI CHIUSURA DELLE CASSE (`CHECK_CLOSING_CASHDESK()`):** LA FUNZIONE DI CHIUSURA DELLE CASSE È IMPLEMENTATA ALL'INTERNO DELLA LIBRERIA STATICA "`LIBSM.C`", CIÒ CHE FA È SCORRERE L'ARRAY "`INFO_QUEUE`" PER CONTARE LE CASSE CHE HANNO AL MASSIMO UN CLIENTE ALL'INTERNO DELLA LORO CODA, TRAMITE L'UTILIZZO DI UN CONTATORE "`COUNT`", SE QUEST'ULTIMO RISULTA ESSERE MAGGIORE O UGUALE DELLA VARIABILE DEFINITA NEL FILE DI CONFIG "`BOUND_MIN`" E C'È ALMENO UNA CASSA DA POTER CHIUDERE VIENE IMPLEMENTATA UNA PROCEDURA DI SCELTA DELLA CASSA. UNA VOLTA CHIUSA LA CASSA E AGGIORNATI I SUOI PARAMETRI VIENE FERMATO IL TEMPO DI APERTURA, TRAMITE UNA CHIAMATA DI `CLOCK_GETTIME()`, E IL TEMPO VIENE INSERITO, SEMPRE IN MUTUA ESCLUSIONE, ALL'INTERNO DELLA LISTA DEI TEMPI DI APERTURA DELLA CASSA. IN SEGUITO ALLA SCELTA E ALLA CHIUSURA DI UNA CASSA, LA FUNZIONE "`CHECK_CLOSING_CASHDESK()`" PROVVEDERÀ AD ESEGUIRE L'OPERAZIONE DI SPOSTAMENTO DEI CLIENTI CHE SI TROVAVANO EVENTUALMENTE SULLA CASSA CHE È STATA APPENA CHIUSA, TALE OPERAZIONE PROVVEDERÀ: ALLA SCELTA DI UNA CASSA APERTA (QUESTA OPERAZIONE NON PREVEDE CHE VENGA SCELTA LA CASSA CHIUSA IN PRECEDENZA), ALLO SPOSTAMENTO DEI CLIENTI NELLA CASSA SCELTA IN PRECEDENZA, AD ESEGUIRE UNA `PTHREAD_COND_SIGNAL` NEL CASO IN CUI LA CASSA SCELTA FOSSE IN ATTESA PER ASSENZA DI CLIENTI NELLA SUA CODA ED INFINE AD AGGIORNARE I PARAMETRI DEI CLIENTI E DELLE CASSE COINVOLTE NELLE PRECEDENTI OPERAZIONI.
- **FUNZIONE DI APERTURA DELLE CASSE (`CHECK_OPEN_CASHDESK()`):** LA FUNZIONE DI APERTURA, COME QUELLA DI CHIUSURA, PROVVEDE ALL'ANALISI DELL'ARRAY CONTENENTE LE LUNGHEZZE DELLE CODE DELLE CASSE. UTILIZZANDO UN CICLO, L'ARRAY "`INFO_QUEUE`", VIENE ANALIZZATO E NON APPENA OCCORRE UNA CASSA CHE RISULTA AVERE IL NUMERO DI CLIENTI IN CODA MINORE DELLA VARIABILE DEFINITA NEL FILE DI CONFIG "`BOUND_MAX`" IL CICLO TERMINA E LA FUNZIONE PROCEDE CON L'IMPLEMENTAZIONE DELLA FUNZIONE DI APERTURA DI UNA CASSA. VIENE CONTROLLATO LO STATO DEL SUPERMERCATO E CHE SIA STATA EFFETTIVAMENTE TROVATA UNA CASSA CHE SODDISFA I REQUISITI PER L'APERTURA DI UNA SECONDA CASSA, SE IL CONTROLLO VIENE SODDISFATTO, LA FUNZIONE "`CHECK_OPEN_CASHDESK()`" IMPLEMENTA UNA FUNZIONE DI SCELTA DI UNA CASSA CHIUSA DA POTER APRIRE. UNA VOLTA SCELTA LA CASSA DA APRIRE SI INIZIA A MISURARE IL SUO TEMPO DI APERTURA TRAMITE UNA CHIAMATA DELLA FUNZIONE `CLOCK_GETTIME()` E SI PROVVEDE AD AGGIORNARE LA VARIABILE "`OPEN`" CHE CONTA IL NUMERO DI CASSE APERTE.

## ANALISI.SH

FUNZIONAMENTO:

- ASPETTA CHE IL PROGRAMMA PRINCIPALE TERMINI CONTROLLANDO IN UN CICLO LA VARIABILE "`PID`" INIZIALIZZATA TRAMITE IL COMANDO "`PIDOF`" ED ESEGUENDO UNA SLEEP DI UN SECONDO.
- ESEGUE IL PARSING DEL FILE DI LOG IMMAGAZZINANDO I VALORI RACCOLTI IN DUE ARRAY DISTINTI, UNO PER I CLIENTI E UNO PER LE CASSE.
- QUANDO MEMORIZZA IL TEMPO DI PERMANENZA NEL SUPERMERCATO O IL TEMPO IN CODA DEI CLIENTI, UTILIZZA UNA PRECISIONE DI 6 CIFRE. QUESTO PER EVITARE CHE IL TEMPO STAMPATO DALLO SCRIPT, ESSENDO MOLTO PICCOLO, RISULTI ESSERE UGUALE A 0