



# JUEGO DELIVERY

## Trabajo Práctico 2

---

### PROGRAMACIÓN CONCURRENTES

BOZZANO, Felipe - Legajo: 102050

OLMOS, Francisco - Legajo: 102369

Profesor: Maximiliano Andrés Eschoyez

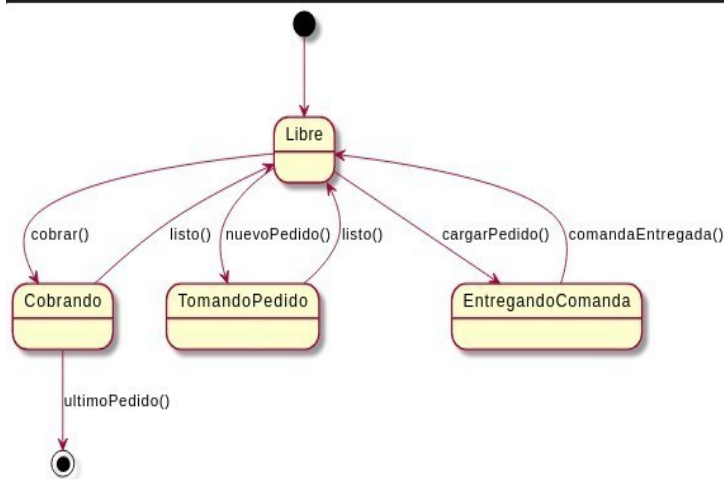
## Problemática:

El objetivo de este proyecto es realizar un controlador de un juego de Delivery por medio de variables compartidas y sincronización entre procesos. El programa deberá manejar 3 procesos diferentes con sus correspondientes hilos. El proceso principal debe ser el que controla el sistema completo, y los otros dos son los productores y consumidores.

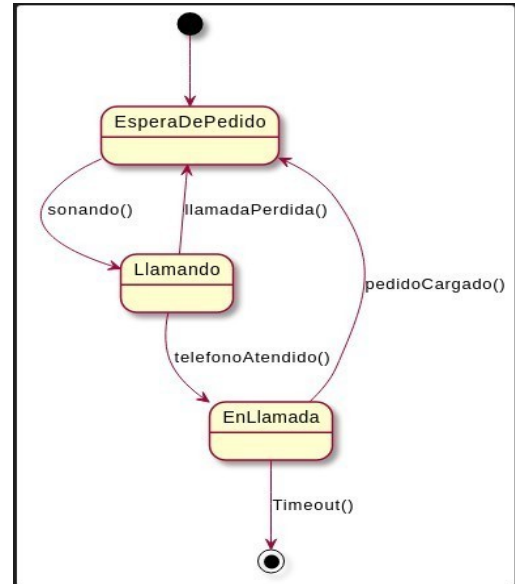
Debe haber un Teléfono que “suene” aleatoriamente por un tiempo determinado el cual debe ser atendido por el Encargado del local. En este caso el Encargado tomará el pedido que recibe del Teléfono y lo cargará junto al resto de pedidos por cocinar, sumando así un punto al “score” del jugador. Llegado el caso de que no pueda atender el Teléfono, entonces este pedido se perderá y no sumará puntos. Por otro lado los cocineros deberán revisar si hay pedidos por cocinar y cocinarlos. Una vez cocinados, los deberán colocar junto al resto de pedidos cocinados. Por último, los Delivery serán los encargados de repartir los pedidos listos y entregar el dinero al Encargado cuando éste esté disponible para cobrarles.

## UML:

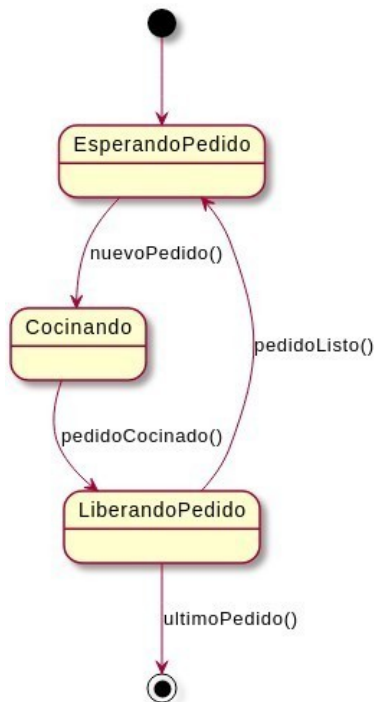
### Encargado



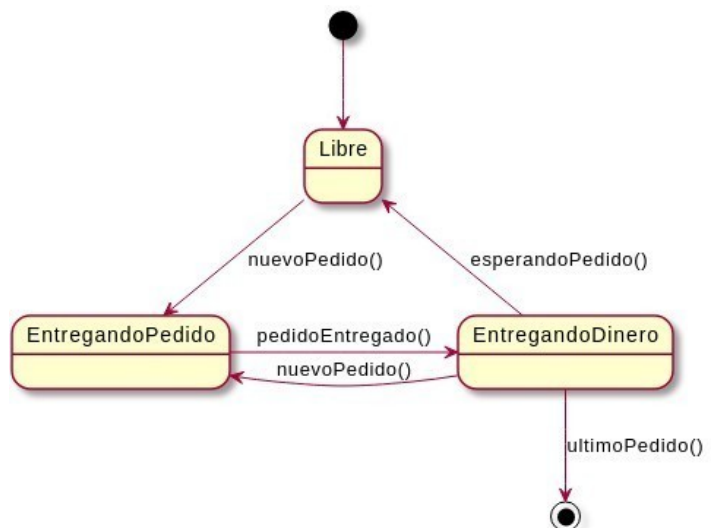
### Teléfono



### Cocinero



### Delivery



## Solución:

Para la implementación del programa contamos con un proceso Teléfono un proceso Encargado, un proceso Cocinero con tres hilos y un proceso Delivery con dos hilos.

El proceso del Encargado es el responsable de atender el Teléfono, pasar los pedidos a los Cocineros y cobrar a los Delivery. Es el actor principal del juego ya que se encarga de la coordinación entre los demás actores y de avisarles cuando tienen que dejar de hacer sus Tareas. Es controlado por el jugador.

El Teléfono implementa una alarma mediante señales IPC que le permite saber cuando tiene que mandar el último pedido que es falso para avisarle al Encargado que debe terminar el juego.

Para la comunicación entre el Encargado y los tres Cocineros implementamos una cola de mensajes, en donde dejamos los pedidos que el Teléfono le manda al Encargado por medio de un pipe. Los cocineros leen de la cola de mensajes y cocinan. Una vez estén cocinados, escriben en otra cola de mensajes que comparten con los Delivery, de donde estos últimos leen para repartirlos.

Los Delivery una vez que entregan los pedidos, depositan el número de pedido entregado y su dinero en una FIFO compartida con el Encargado.

El Encargado lee de esta FIFO y cobra los pedidos que fueron escritos por los Delivery en el orden en el que entraron.

## INSTRUCCIONES DE JUEGO:

El programa cuenta con un Menú de opciones que le permiten al usuario comenzar el juego apretando la tecla "1", ver su puntuación con "2" o salir apretando "3".

Para poder sumar puntos solo se debe atender el teléfono cuando este suene, lo cual se indicará con un mensaje en rojo diciendo "teléfono sonando", apretando la tecla "a" y luego "ENTER". No se puede atender el teléfono si se tiene una comanda en la mano. Esto quiere decir que para poder atender nuevamente el teléfono se debe pasar el pedido al cocinero antes de atender otro llamado.

En caso de no atenderlo en el lapso de 1 segundo, se perderá el llamado y la posibilidad de incrementar la puntuación del jugador.

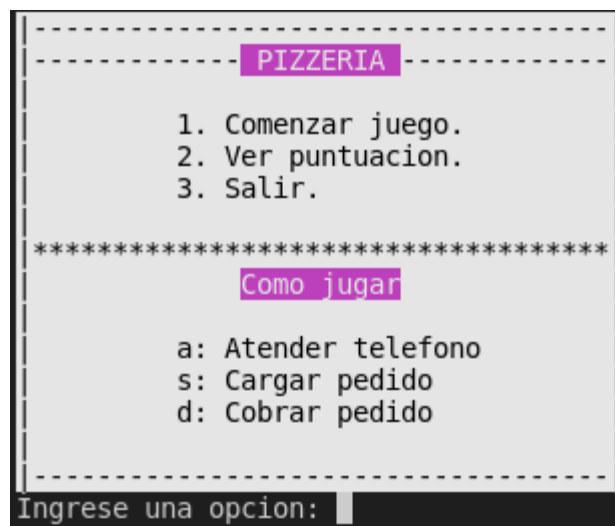
El siguiente paso es cargar el pedido recibido desde el teléfono a los cocineros. Para ello se debe apretar la tecla “s” y luego “ENTER” cuando se le indique por pantalla un mensaje en color magenta indicando “comanda de pedido X lista para cargar”.

Es importante también cobrarle a los Delivery. Para poder hacer esto se debe apretar la tecla “d” y luego “ENTER” cuando se le indique por pantalla un mensaje en color amarillo diciendo “dejando dinero en caja de pedido tanto”.

Una vez se hayan cobrado todos los pedidos atendidos cocinados y entregados se deberá finalizar al juego apretando la letra “d” y luego “ENTER”.

**NOTA:** Es importante aclarar que si se aprieta la tecla “s” y luego “ENTER” sin antes atender el teléfono es decir sin una comanda en la mano, entonces se mostrara un cartel en blanco con lo siguiente: “Atender el teléfono antes de cargar un pedido”

El menú de opciones se verá de la siguiente forma:



Una ejecución del programa se verá de la siguiente forma:

```
mq_open_encargadoCocineros_enc_ok()
mq_open_encargadoCocineros_coc_ok()
mq_open_cocinerosDelivery_coc_ok()
mq_open_cocinerosDelivery_del_ok()
FIFO_open_deliveryEncargado_ok()
sem_open_semTelefono_tel_ok()
-----
a      telefono sonando
      comanda de pedido 1 lista para cargar
s      telefono sonando
a      Se perdio la llamada
s
      telefono sonando
      comanda de pedido 2 lista para cargar
s
      atender el telefono antes de cargar un pedido
      Pedido 1 listo para cobrar
d      telefono sonando
a      $200 guardados de pedido 1
s      Se perdio la llamada
      telefono sonando
      comanda de pedido 4 lista para cargar
      Pedido 2 listo para cobrar
d      telefono sonando
a      $300 guardados de pedido 2
      comanda de pedido 0 lista para cargar
s      telefono sonando
      Pedido 4 listo para cobrar
d      Se perdio la llamada
```

```

telefono sonando
comanda de pedido 4 lista para cargar
Pedido 2 listo para cobrar
d telefono sonando
$300 guardados de pedido 2
a comanda de pedido 0 lista para cargar
telefono sonando
Pedido 4 listo para cobrar
d Se perdio la llamada
Dueño llamando para cerrar local
$500 guardados de pedido 4
a avisar a los cocineros que cierren la cocina
s Pedido 0 listo para cobrar
Presione d para cerrar el local
d $100 guardados de pedido 0
d Cerrando local

```

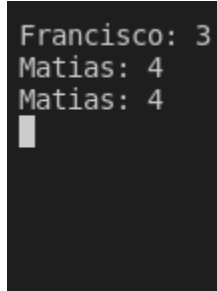
```

-----
sem_close_semTelefono_enc_ok()
sem_unlink_semTelefono_enc_ok()
mq_close_encargadoCocineros_coc_ok()
mq_close_encargadoCocineros_enc_ok()
mq_unlink_encargadoCocineros_enc_ok()
mq_close_cocinerosDelivery_del_ok()
mq_close_cocinerosDelivery_coc_ok()
mq_unlink_cocinerosDelivery_coc_ok()
FIFO_unlink_deliveryEncargado_ok()
-----

```

Escriba su nombre: Jugador1

Al apretar la letra “d” al final del juego cuando se le indica, deberá escribir su nombre con el que quiere que se le registre la puntuación obtenida. El programa genera un archivo de texto “puntuacion.txt” con las puntuaciones de los jugadores.



```
Francisco: 3
Matias: 4
Matias: 4
█
```

Como compilar el programa:

```
gcc -c ./TP-2/DeleteCreate.c -o DeleteCreate.o
```

```
gcc -pthread -Wall -pedantic ./TP-2/delivery_TP2.c DeleteCreate.o -o
prueba.c -lrt
```

Como ejecutar el programa:

```
./prueba.c
```



## Conclusión:

Al igual que los hilos, el correcto manejo de los procesos garantizan la sincronización del juego. La diferencia central es que no comparten el mismo espacio de ejecución, por lo que se necesitan herramientas para comunicar los mensajes entre los mismos. Estas herramientas, combinadas con las vistas anteriormente (Semáforos, Mutex, etc) aseguran la sincronización entre procesos en este juego.

Dividir el código y comentarlo permitió que el paso de hilos a procesos fuese bastante simple, y la utilización de bibliotecas nos limpia el código principal y ayuda a separar la sincronización de la implementación del juego.