

# Primer Proyecto: Restorrente - El restaurant concurrente

75.59 - Técnicas de Programación Concurrente I

---

## Objetivo

El objetivo de este proyecto consiste en implementar la simulación parcial del funcionamiento de un restaurant.

## Requerimientos Funcionales

Los requerimientos funcionales son los siguientes:

1. Al restaurant ingresan grupos de comensales que son recibidos en la puerta por los recepcionistas.
2. Cada recepcionista atiende a un grupo de personas a la vez y los ubica en una mesa disponible. Si no hay más mesas disponibles, los hace pasar al living del restaurant, donde esperarán a que se desocupe una mesa.
3. Si todos los recepcionistas están ocupados, las personas esperan en la puerta del restaurant.
4. Cuando las personas están en la mesa, le ordenan al mozo un pedido de comida. El mozo toma nota de lo solicitado y le solicita la comida al cocinero. Hay un sólo cocinero que atiende a los mozos de a uno por vez. El cocinero le entrega el pedido al mozo y éste a los clientes de la mesa, quienes luego comerán.
5. Los clientes pueden repetir el ciclo de ordenar comida y comer, cuantas veces lo deseen.
6. Una vez que terminaron de comer, los clientes le solicitan al mozo "la cuenta". Le pagan el importe al mozo, quien lo deposita en la caja del restaurant. Una vez que pagaron, los clientes se retiran del restaurant.
7. De forma inesperada y sorpresiva, se corta el suministro de energía eléctrica en el restaurant, quedando totalmente a oscuras. Los clientes, furiosos, se van del restaurant sin pagar lo que hayan consumido hasta el momento. También se van los que estaban esperando en el living. Cuando se resuelve el problema eléctrico, el restaurant vuelve a funcionar normalmente, recibiendo nuevos clientes.
8. La simulación finaliza cuando todas las personas que iban a comer ese día en el restaurant se retiraron.
9. Periódicamente, el Gerente del restaurant consulta:
  - a) La cantidad de dinero en la caja.
  - b) La cantidad de dinero que el restaurant perdió de facturar debido a los cortes eléctricos.
  - c) La cantidad de grupos de personas esperando en el living.
10. Los siguientes parámetros deben ser configurables sin necesidad de recompilar el código:
  - a) La cantidad de recepcionistas

- b) La cantidad de mesas
  - c) La cantidad de mozos
  - d) El menú del restaurant: el listado de platos y el precio de cada uno
11. Durante la ejecución de la simulación se deberá poder lanzar manualmente el corte de energía eléctrica.

## Requerimientos no Funcionales

Los siguientes son los requerimientos no funcionales de la aplicación:

1. El proyecto deberá ser desarrollado en lenguaje C o C++, siendo este último el lenguaje de preferencia.
2. La simulación puede no tener interfaz gráfica y ejecutarse en una o varias consolas de línea de comandos.
3. El proyecto deberá funcionar en ambiente Unix / Linux.
4. La aplicación deberá funcionar en una única computadora.
5. El programa deberá poder ejecutarse en "modo debug", lo cual dejará registro de la actividad que realiza en un único archivo de texto para su revisión posterior. Se deberá poder seguir el recorrido de cada uno de los grupos de personas. Se deberá también poder observar el momento en que ingresan y se retiran del restaurante.
6. Las facilidades de IPC que se podrán utilizar para la realización de este proyecto son las que abarcan la primera parte de la materia, es decir, hasta el primer parcial. Dichas facilidades son:
  - a) Memoria compartida
  - b) Señales
  - c) Pipes y fifos
  - d) Locks
  - e) Semáforos

Cualquier otra facilidad queda expresamente excluida para este proyecto.

## Tareas a Realizar

A continuación se listan las tareas a realizar para completar el desarrollo del proyecto:

1. Dividir el proyecto en procesos. El objetivo es lograr que la simulación esté conformada por un conjunto de procesos que sean lo más sencillos posible.
2. Una vez obtenida la división en procesos, establecer un esquema de comunicación entre ellos teniendo en cuenta los requerimientos de la aplicación. ¿Qué procesos se comunican entre sí? ¿Qué datos necesitan compartir para poder trabajar?
3. Tratar de mapear la comunicación entre los procesos a los problemas conocidos de concurrencia.
4. Determinar los mecanismos de concurrencia a utilizar para cada una de las comunicaciones entre procesos que fueron detectadas en el ítem 2. No se requiere la utilización de algún mecanismo específico, la elección en cada caso queda a cargo del grupo y debe estar debidamente justificada.
5. Realizar la codificación de la aplicación. El código fuente debe estar documentado.

## Entrega

La entrega del proyecto comprende lo siguiente:

1. Informe, se deberá presentar impreso en una carpeta o folio y en forma digital (PDF) a través del campus
2. El código fuente de la aplicación, que se entregará únicamente mediante el campus

La entrega en el campus estará habilitada hasta las 19 hs de la fecha indicada oportunamente.

El informe a entregar debe contener los siguientes items:

1. Breve análisis del problema, incluyendo una especificación de los casos de uso de la aplicación.
2. Detalle de resolución de la lista de tareas anterior.
3. Diagrama que refleje los procesos, el flujo de comunicación entre ellos y los datos que intercambian.
4. Diagramas de clases realizados.
5. Diagrama de transición de estados de un mozo.