

Proyecto 2: Reservaciones de hotel utilizando hilos y procesos

Situación a Modelar:

- Este programa simula un sistema de reservas en un hotel, donde los clientes pueden reservar habitaciones en línea. El hotel tiene varias habitaciones, y el objetivo es evitar problemas como overbooking y conflictos de acceso a la información de disponibilidad.

Consecuencias Nocivas de la Concurrency:

- Overbooking: Si múltiples clientes intentan reservar la misma habitación al mismo tiempo, podríamos tener overbooking, lo que resultaría en problemas para los huéspedes y pérdida de ingresos para el hotel.
- Conflictos de Acceso: Si varios clientes intentan acceder simultáneamente a la información de disponibilidad, podríamos tener inconsistencias en los datos y mostrar información incorrecta.

Eventos Concurrentes a Controlar:

- Reservas: Varios clientes intentan reservar habitaciones al mismo tiempo.
- Verificación de Disponibilidad: Varios clientes consultan la disponibilidad de habitaciones simultáneamente.

Eventos Concurrentes sin Importancia en el Ordenamiento:

- En este caso, el orden relativo de los eventos no es crítico, ya que el sistema debe manejar las reservas y las verificaciones de disponibilidad de manera concurrente sin depender del orden específico en que ocurren.

Documentación del Código:

Mecanismos de Sincronización Empleados:

- Mutex (Mutex de habitación): Utilizado para garantizar la exclusión mutua al reservar habitaciones.
- Semáforo (Semáforo de disponibilidad): Utilizado para controlar el acceso concurrente a la información de disponibilidad.

Lógica de Operación:

- Reservas: Los clientes adquieren el semáforo de disponibilidad, verifican la disponibilidad, adquieren el mutex correspondiente para la habitación, actualizan el estado de la habitación y liberan el semáforo y el mutex.
- Verificación de Disponibilidad: Los clientes adquieren el semáforo de disponibilidad, consultan la disponibilidad y liberan el semáforo.

Estado Compartido:

- Variables Globales: La variable habitaciones representa el estado de cada habitación (disponible o reservada).
- Descripción del Avance de Hilos/Procesos:
- Reservas: Verificación de disponibilidad, adquisición de semáforo, adquisición de mutex, actualización del estado de la habitación, liberación de semáforo y mutex.
- Verificación de Disponibilidad: Adquisición de semáforo, consulta de disponibilidad, liberación de semáforo.

Descripción de la Interacción:

- Reservas y Disponibilidad: Sincronización mediante el uso de mutex y semáforo para garantizar consistencia y evitar conflictos.

Entorno de Desarrollo:

- Lenguaje: Python 3.8
- Bibliotecas: threading (estándar en Python)
- Sistema Operativo: Windows 10/11

Instrucciones de Ejecución:

- Asegúrate de tener Python 3.8 instalado en tu sistema.
- Ejecuta el archivo en la terminal o consola de comandos.

```
PS C:\Users\juanp> & C:/Users/juanp/
Reservando habitacion1
Reserva de habitacion1 completada.
Reservando habitacion2
Reserva de habitacion2 completada.
```

En la ejecución se toman 2 habitaciones, pero puede ser escalable el numero de habitaciones ya que el mutex se encargara de bloquear las reservaciones para que no existe discrepancia entre los usuarios que reservan.