

Laboratorio 8

1.

Aspecto	Mutex (Exclusión Mutua)	Variables de Condición	Semáforos	Barreras
Función principal	Controlar el acceso a un recurso compartido.	Coordina la ejecución de hilos en función de condiciones.	Controlar el acceso a recursos o coordinar hilos.	Sincronizar la ejecución de un grupo de hilos.
Cuándo se usa	Cuando varios hilos necesitan acceso exclusivo a un recurso compartido para evitar condiciones de carrera.	Cuando se necesita que un hilo espere a que se cumpla una condición antes de continuar.	Cuando se necesita limitar el acceso a un recurso (semáforo contador) o asegurar la exclusión mutua (semáforo binario).	Cuando se desea sincronizar un conjunto de hilos para que continúen al mismo tiempo después de alcanzar un punto común
Sincronización	Exclusión mutua.	Sincronización basada en condiciones.	Exclusión mutua (semáforo binario) o sincronización (semáforo contador).	Sincronización de hilos.
Número de hilos involucrados	Un solo hilo tiene acceso a la vez.	Puede coordinar a múltiples hilos, pero solo uno se despierta por señal.	Varios hilos pueden interactuar; el semáforo puede contar varios accesos (semáforo contador).	Todos los hilos deben llegar al punto de sincronización.
Bloqueo/Espera	Un hilo se bloquea si el mutex está ocupado.	Los hilos pueden esperar hasta que se cumpla la condición.	Los hilos pueden bloquearse si el semáforo no está disponible (valor ≤ 0).	Los hilos se bloquean hasta que todos alcancen la barrera.
Reinicialización	No requiere reinicialización explícita; se libera y adquiere repetidamente.	Debe volver a evaluarse la condición después de cada uso.	Puede reiniciarse al incrementar o decrementar el contador (dependiendo del tipo de semáforo).	Necesario reiniciarla para reutilizarla en otra ronda de sincronización.

Ejercicio 02

Responde:

1. ¿Cuál es el recurso compartido y cómo podría afectarse por las condiciones de carrera? El recurso compartido es el saldo de la cuenta bancaria. Podría afectarse por las condiciones de carrera si varios clientes intentan retirar dinero al mismo tiempo, ya que el saldo podría quedar en un estado inconsistente.
2. ¿Qué pasa si no se utiliza ninguna forma de sincronización al acceder al recurso compartido? Si no se utiliza ninguna forma de sincronización al acceder al recurso compartido, varios clientes podrían retirar dinero al mismo tiempo, lo que podría causar que el saldo quede en un estado inconsistente o que se realicen retiros sin saldo suficiente.
3. ¿Cómo sabrás si un cliente tiene suficiente saldo antes de hacer un retiro? ¿Cuál mecanismo de control implementarías? Para saber si un cliente tiene suficiente saldo antes de hacer un retiro, se puede implementar un mecanismo de control que verifique si el monto a retirar es menor o igual al saldo disponible. Se puede utilizar un semáforo para controlar el acceso al cajero y garantizar que solo un cliente pueda retirar dinero a la vez.
4. Si utilizas semáforo, ¿cuál es el valor inicial del semáforo y por qué? El valor inicial del semáforo es 1, ya que se desea que solo un cliente pueda acceder al cajero a la vez. Al tener un valor inicial de 1, se garantiza que solo un hilo pueda entrar en la sección crítica a la vez.

Ejercicio 03

Responde:

1. Explica el funcionamiento del sistema ¿Se inician y finalizan correctamente los procesos? ¿Por qué?
El programa consiste en un sistema de productores y consumidores en el que varios productores fabrican piezas de sillas y los consumidores las ensamblan para formar sillas completas. Los procesos no se inician y finalizan correctamente, porque no se usan semáforos, en su lugar se usa un bucle while(true) dentro de cada función de productor y consumidor. De esta manera puede pasar que el productor intente agregar una pieza del buffer lleno o el consumidor intente retirar una pieza del buffer vacío.
2. ¿Qué modificaciones deben realizarse para que los productores dejen de producir cuando se alcance el límite de sillas, y los consumidores también terminen?
Se necesita agregar una condición en los productores para que dejen de fabricar cuando se alcance el límite máximo de sillas.
Los consumidores deben ser capaces de detectar que no habrá más piezas disponibles y finalizar correctamente cuando se alcance el límite de producción.
Usar semáforos para sincronizar los accesos al buffer y así evitar que los hilos accedan al mismo tiempo.
3. Agrega el código necesario para que se genere un reporte antes de finalizar la ejecución. El reporte debe indicar cuántas sillas se fabricaron en totalidad, y cuántas piezas de cada tipo sobraron (quedarán en almacén).