Universidad de Granada

GRADO EN INGENIERÍA INFORMÁTICA

Prácticas
Sistemas Concurrentes y Distribuidos

Práctica 1. Sincronización de hebras con semáforos

Ruiz Adán, Francisco

Granada, octubre de 2020

1. Solución al problema del productor-consumidor

a. Solución LIFO

Para esta solución, se han empleado cuatro variables:

1) Un semáforo, **libres**, inicializado al tamaño del buffer donde se guardan los datos. Esta variable la utilizará el productor para saber si debe o no, esperar para insertar un nuevo dato en el buffer.
    i. El productor usará sem_wait(), cuando produzca e inserte un nuevo valor en el buffer. Si el valor del semáforo es 0, el productor deberá esperar, ya que esto significa que el buffer estaría lleno.
    ii. El consumidor usará sem_signal() cuando lea un valor del buffer.

2) Un semáforo, **ocupadas**, inicializado a 0. Esta variable la utilizará el consumidor para saber si hay algún dato en el buffer para consumir.
    i. El consumidor usará sem_wait(), antes de hacer nada para comprobar que hay algún valor que consumir en el buffer. Si el valor del semáforo es 0, el consumidor deberá esperar, ya que esto significa que el buffer estaría vacío.
    ii. El productor usará sem_signal() sobre el semáforo justo después de insertar un nuevo valor en el buffer.

3) Un entero, **primera_libre**, que indica la posición libre en el buffer. El productor, insertará un nuevo dato en el lugar indicado por esta variable y más tarde incrementará su valor. Por otro lado, el consumidor leerá el dato situado a primera_libre – 1 del buffer y decrementará su valor. Como al inicio suponemos que el buffer estará vacío, la inicializaremos a 0.

4) Una variable tipo mutex, **mtx**, para exclusión mutua, ya que ambas hebras modifican primera_libre

5) Un **vector de enteros**, con un tamaño determinado donde se irán escribiendo/leyendo los valores producidos.

b. Solución FIFO

Para esta solución se emplean las mismas variables que la solución LIFO, pero se añaden tres cambios:

i. Se añade una nueva variable, **primera_ocupada**, que indica la primera posición ocupada en el buffer. Inicialmente valdrá 0. Aunque inicialicemos esta variable a 0 y el buffer esté vacío no se producirá ningún error, ya que inicialmente, el consumidor deberá esperar a que haya algún valor en el buffer para poder leer.

ii. En la solución LIFO, tanto el productor como el consumidor hacían uso de la variable **primera_libre**. Ahora, esta variable solamente la usará el productor para saber la siguiente posición libre del buffer. Por otro lado, el consumidor hará uso de la variable **primera_ocupada** para leer la primera posición ocupada del buffer.

iii. Se prescinde de la variable tipo mutex ya que ambas hebras modifican diferentes variables.

2. Solución al problema del productor-consumidor

Para este ejercicio se han creado un semáforo, **puede_producir**, que usará el estanquero y un vector de semáforos, **puede_retirar**, de tamaño el número de fumadores.

Ya que inicialmente ningún fumador tendrá su ingrediente, el semáforo **puede_pruducir** estará inicializado a 1, indicando que el estanquero al inicio puede producir un ingrediente. El vector de semáforos, estará inicialmente inicializado a 0.

El semáforo **puede_producir** nos servirá para indicar al estanquero que puede producir un nuevo ingrediente.

El vector de semáforos nos servirá para saber que fumador puede continuar su ejecución dependiendo del ingrediente que haya puesto el estanquero.

La hebra estanquero, ejecutará al inicio sem_wait() sobre su semáforo para saber si debe o no producir un nuevo ingrediente. Si el estanquero ha producido un nuevo ingrediente, al final, ejecutará sem_signal() sobre el semáforo del fumador asociado a ese ingrediente.

Las hebras de los fumadores, inicialmente ejecutarán sem_wait() sobre su correspondiente semáforo, para esperar a que el estanquero produzca el ingrediente que necesitan. Si la hebra de un fumador reanuda su ejecución debido a la notificación del estanquero, cogerá el ingrediente y ejecutará sem_signal() sobre el semáforo del estanquero para notificarle que puede poner otro ingrediente.