

Mutex

Abstract

A continuación se presenta un análisis correspondiente a lo requerido en el taller de "Mutex" asignado en la clase de sistemas operativos 2019-I

1. Explicación del código dado

Tras realizar la transcripción y ejecución del código se puede observar que el funcionamiento del mismo corresponde a un cierto número N de hilos los cuales se encargan de realizar la impresión del valor asignado a *data*, $p+1$ veces. Con lo anteriormente mencionado y sin ningún método de sincronización de procesos se tiene el problema de que la impresión se da de manera desorganizada, esto es, que durante la impresión realizada por el hilo A se ven filtrados valores impresos por el procesamiento de otro hilo B.

2. Implementación y análisis del Mutex

Para la implementación del método de sincronización *Mutex* se identificaron las zonas críticas, estas corresponden dentro de la función *process* a donde se inicia la impresión de los valores y donde se finaliza la misma. En estos se hace uso de las funciones características de *Mutex*, *pthread_mutex_lock* y *pthread_mutex_unlock* respectivamente, las cuales reciben como parámetro un puntero al valor del mutex, inicializado en el *main* haciendo uso de *pthread_mutex_init* y destruido luego de su uso con *pthread_mutex_destroy*.

Una vez realizada la implementación del mutex dentro del código dado en un principio por el enunciado del taller, se obtiene que la impresión de los valores de

cada hilo se ve sincronizada, esto es, que cuando un hilo se encuentra en procesamiento este no será interrumpido por la impresión de algún otro hilo, sino que los demás hilos esperaran hasta que el buffer de impresión no esté bloqueado para ellos proceder a la impresión de los datos. Analizando un poco más en detalle, cuando cada hilo entra en la función de *process*, justo antes de comenzar la impresión de los valores, mediante *pthread_mutex_lock* se bloquea se notifica a los demás hilos que el buffer de impresión está en uso y no puede ser interrumpido, tras finalizar se desbloquea el buffer notificando con el mutex mediante *pthread_mutex_unlock*.