

Ejercicios sobre sincronización y paralelización por tareas

Todos los ejercicios propuestos aparecen en el libro "The little book of semaphores".

Problema de Santa Claus (página #137)

Santa Claus duerme en su tienda en el polo norte y sólo puede ser despertado cuando se uno de dos eventos:

1. han regresado todos los nueve renos de sus vacaciones por las playas del Pacífico Sur, o
2. algunos de los duendes tienen dificultades haciendo juguetes y Santa debe ayudarles.

Para permitirle a Santa descansar, los eventos anteriores están sujetos a las siguientes restricciones para despertarlo:

1. sólo cuando se agrupan tres duendes con problemas Santa los atiende,
2. si algún otro duende llega cuando Santa atiende a otros tres, el nuevo duende debe esperar a que los otros tres terminen (y que se complete otro grupo de tres duendes),
3. si Santa encuentra a tres duendes en su puerta esperando junto al último reno que ha regresado de la playa, los duendes tendrán que esperar hasta pasada la navidad porque en ese momento es más importante acondicionar su trineo,
4. se supone que los renos llegan y esperan en una cálida cabaña hasta que el último llegue, en lugar de querer fugarse a la playa de nuevo; al llegar el último son enganchados al trineo,
5. Santa atiende todos los duendes que puede antes de partir con los renos,
6. hay exactamente nueve renos, pero una cantidad indefinida de duendes.

Elabore un programa con OpenMP / C++ que simule la tienda de Santa, los 9 renos y N duendes, valor suministrado por el usuario. Asegúrese de identificar correctamente los recursos compartidos y que se creen los hilos requeridos. Asegúrese que su programa no produce "deadlock" entre los hilos que compiten por algún recurso compartido. Su programa deberá generar un archivo de texto que muestre los diferentes eventos de la simulación en el orden correcto, una bitácora de lo acontecido en la simulación.

Problema de la barbería (página #121)

El problema original de la barbería fue propuesto por Dijkstra. Una variación del mismo la plantearon Silberschatz y Galvin en "Operating Systems Concepts".

Tenemos una barbería con una sala de espera con N sillas, y el cuarto donde se ubica la silla del barbero. Si no hay clientes para ser atendidos, el barbero se duerme. Si un cliente entra a la barbería y todas las sillas están ocupadas, simplemente se va. Si el barbero está ocupado, pero hay sillas disponibles, el cliente recién llegado ocupa una de las sillas libres. Si el barbero está dormido, el cliente recién llegado lo despierta para que lo atienda.

Elabore un programa con OpenMP / C++ que simule la barbería, con N la cantidad de sillas como un valor suministrado por el usuario. Asegúrese de identificar correctamente los recursos compartidos y que se creen los hilos requeridos. Asegúrese que su programa no produce "deadlock" entre los hilos que compiten por algún recurso compartido. Su programa deberá generar un archivo de texto que muestre los diferentes eventos de la simulación en el orden correcto, una bitácora de lo acontecido en la simulación.