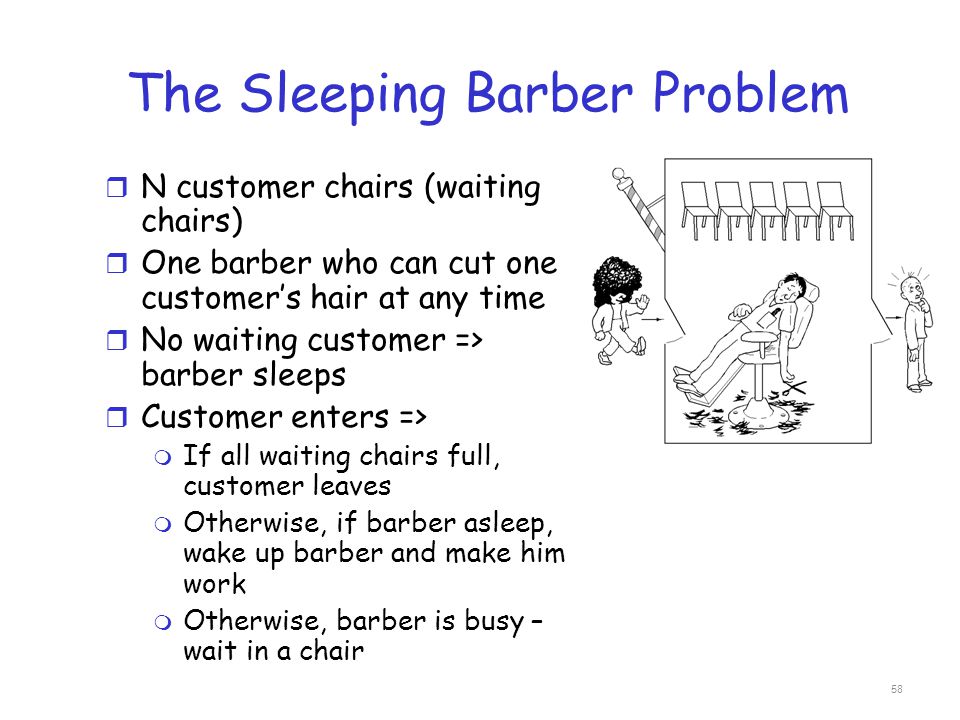
The Sleeping Barber Problem: A Classical concurrent programming Building

Widely used in Operative System



Hay población de clientes que necesitan afeitarse de vez en cuando, siendo el medio tiempo de no quererse afeitarse Taway

Para afeitarse cada cliente se acerca a la barbería y revisa si el barbero está libre o si hay asientos de espera vacíos. Existen cinco asientos 0,1,2,3,4, donde 0 es la silla del barbero y 1,2,3,4 sillas de espera.

Si no hay clientes en la silla 0 el barbero se duerme.

Con un cliente en la silla 0 el barbero se despierta y atiende al cliente por un tiempo de servicio Tserv, (por ejemplo 10 ciclos).

Si el barbero está ocupado y llegan más cliente se forma una cola en las sillas 1,2,…por orden de llegada (Queue) y se atiende según un orden FIFO.

Cuando el barbero termina con un cliente lo despacha y toda la cola se desplaza (shift) hacia la silla 0, donde el que llega atendido por el barbero, etc.

Si llega un cliente y la cola está llena se marcha, si hay sillas vacías se sienta y espera.

En el programa adjunto se presenta la solución en C++. Esta solución es parcial y aún queda por resolver el acceso a la zona crítica (sillas del barbero) en términos de threads. Por favor compílenlo, estúdienlo, ordenen los mensajes y agreguen otros mensajes al sistema, como por ejemplo:

printf("Barber Sleeping\n");

printf("Barber woke up\n");

En la próxima práctica lo terminamos.