**6.1. Cite ejemplos de recursos reutilizables y consumibles.**

Primero que nada se puede definir el **interbloqueo** como el bloqueo *permanente* de un conjunto de procesos que o bien compiten por recursos del sistema o se comunican entre sí. Un conjunto de procesos está interbloqueado cuando cada proceso del conjunto está bloqueado esperando un evento (normalmente la liberación de algún recurso requerido) que sólo puede generar otro proceso bloqueado del conjunto. El interbloqueo es permanente porque no puede producirse ninguno de los eventos. A diferencia de otros problemas que aparecen en la gestión de procesos concurrentes, no hay una solución eficiente para el caso general.

Un **recurso reutilizable** es aquél que sólo lo puede utilizar de forma segura un proceso en cada momento y que no se destruye después de su uso. Los procesos obtienen unidades del recurso que más tarde liberarán para que puedan volver a usarlas otros procesos. Algunos ejemplos de recursos reutilizables incluyen procesadores, canales de E/S, memoria principal y secundaria, dispositivos, y estructuras de datos como ficheros, bases de datos y semáforos.

Un **recurso consumible** es aquél que puede crearse (producirse) y destruirse (consumirse). Normalmente, no hay límite en el número de recursos consumibles de un determinado tipo. Un proceso productor desbloqueado puede crear un número ilimitado de estos recursos. Cuando un proceso consumidor adquiere un recurso, el recurso deja de existir. Algunos ejemplos de recursos consumibles son las interrupciones, las señales, los mensajes y la información en *buffers* de E/S.

**6.2. ¿Cuáles son las tres condiciones que deben cumplirse para que sea posible un interbloqueo?**

Deben presentarse tres condiciones de gestión para que sea posible un interbloqueo:

1. **Exclusión mutua.** Sólo un proceso puede utilizar un recurso en cada momento. Ningún proceso puede acceder a una unidad de un recurso que se ha asignado a otro proceso.
2. **Retención y espera.** Un proceso puede mantener los recursos asignados mientras espera la asignación de otros recursos.
3. **Sin expropiación.** No se puede forzar la expropiación de un recurso a un proceso que lo posee.

**6.3. ¿Cuáles son las cuatro condiciones que producen un interbloqueo?**

Si se cumplen estas tres condiciones se puede producir un interbloqueo, pero aunque se cumplan puede que no lo haya. Para que realmente se produzca el interbloqueo, se requiere una cuarta condición:

1. **Espera circular.** Existe una lista cerrada de procesos, de tal manera que cada proceso posee al menos un recurso necesitado por el siguiente proceso de la lista.

**6.4. ¿Cómo se puede prever la condición de retención y espera?**

La condición de retención y espera puede eliminarse estableciendo que un proceso debe solicitar al mismo tiempo todos sus recursos requeridos, bloqueándolo hasta que se le puedan conceder simultáneamente todas las peticiones. Esta estrategia es insuficiente en dos maneras. En primer lugar, un proceso puede quedarse esperando mucho tiempo hasta que todas sus solicitudes de recursos puedan satisfacerse, cuando, de hecho, podría haber continuado con solamente algunos de los recursos. En segundo lugar, los recursos asignados a un proceso pueden permanecer inutilizados durante un periodo de tiempo considerable, durante el cual se impide su uso a otros procesos. Otro problema es que un proceso puede no conocer por anticipado todos los recursos que requerirá.

Hay también un problema práctico creado por el uso de una programación modular o una estructura multihilo en una aplicación. La aplicación necesitaría ser consciente de todos los recursos que se solicitarán en todos los niveles o en todos los módulos para hacer una solicitud simultánea.

**6.5. Enumere dos maneras cómo se puede prever la condición de sin expropiación.**

Esta condición se puede impedir de varias maneras. En primer lugar, si a un proceso que mantiene varios recursos se le deniega una petición posterior, ese proceso deberá liberar sus recursos originales y, si es necesario, los solicitará de nuevo junto con el recurso adicional. Alternativamente, si un proceso solicita un recurso que otro proceso mantiene actualmente, el sistema operativo puede expropiar al segundo proceso y obligarle a liberar sus recursos. Este último esquema impediría el interbloqueo sólo si no hay dos procesos que posean la misma prioridad.

Esta estrategia es sólo práctica cuando se aplica a recursos cuyo estado se puede salvar y restaurar más tarde, como es el caso de un procesador

**6.6. ¿Cómo se puede prever la condición de espera circular?**

La condición de espera circular se puede impedir definiendo un orden lineal entre los distintos tipos de recursos. Si a un proceso le han asignado recursos de tipo *R*, posteriormente puede pedir sólo aquellos recursos cuyo tipo tenga un orden posterior al de *R*.

**6.7. ¿Cuál es la diferencia entre predicción, detección y prevención del interbloqueo?**

Existen tres estrategias para el tratamiento del interbloqueo. En primer lugar, se puede prevenir el interbloqueo adoptando una política que elimine una de las condiciones (las 4 condiciones enumeradas previamente). En segundo lugar, se puede predecir el interbloqueo tomando las apropiadas decisiones dinámicas basadas en el estado actual de asignación de recursos. En tercer lugar, se puede intentar **detectar** la presencia del interbloqueo (se cumplen las 4 condiciones) y realizar las acciones pertinentes para recuperarse del mismo.