

(1) ¿Cuál es la principal ventaja de la multiprogramación? ¿Que problemas nuevos le acarrea el sistema operativo?

La multiprogramación permite la ejecución de programas de forma concurrente incrementando la utilización de los recursos del sistema (p.ej. el procesador). Algunos de los problemas que acarrea son:

- Concurrencia en el acceso a los recursos (e.g. almacenamiento secundario, periféricos, etc.)
- Asignación justa de los recursos a los procesos. Se debe contar con técnicas de planificación de CPU, de almacenamiento secundario, etc.

(2) ¿Para qué sirve la protección de CPU? ¿Cómo se implementa?

Sirve para que un proceso no acapare el procesador. Para implementarse se utiliza un timer en hardware y una rutina de atención a las interrupciones del timer. La instrucción que carga el contador del timer debe ser privilegiada.

(3) Describa las acciones del sistema operativo cuando ocurre un cambio de contexto:

a. entre hilos.

b. entre procesos.

a.

- Salvar el estado actual del hilo en ejecución. Esto implica salvar el estado de algunos registros del CPU, por ejemplo: el puntero al stack, el puntero a instrucción, etc.
- Cambiar el estado del hilo que estaba ejecutando de ejecutando al estado que corresponda.
- Cambiar el estado del hilo nuevo a ejecutando
- Cargar el estado del nuevo hilo (puntero de stack, IP, etc.)

b.

- Salvar el estado del proceso que está ejecutando en su PCB. Esto implica salvar el estado de todos los registros del CPU.
- Cambiar el estado del proceso que estaba ejecutando al que corresponda.
- Cambiar el estado del proceso nuevo a ejecutando.
- Cargar el estado del nuevo proceso a los registros del CPU a partir de su PCB.

(4) Mencione cuáles son las tres estrategias que utilizan los SO modernos para enfrentarse a un deadlock.

1) Prevenir o evitar: utilizar un protocolo para prevenir o evitar que ocurra un deadlock. Por ejemplo: para prevenir deadlocks es necesario asegurar que al menos una de las 4 condiciones necesarias nunca se cumpla (exclusión mutua, retención y espera, no expropiable, espera circular); para evitar deadlocks se puede utilizar algoritmo del banquero.

2) Detectar y recuperarse: permitir que ocurran deadlocks pero detectarlos y recuperarse. Para detectar un deadlock puede usarse una técnica de detección de ciclos en grafo de asignación de recursos o una variante del Algoritmo del Banquero. Para recuperarse es necesario tomar acciones como matar procesos.

3) Desentenderse: ignorar el problema (mirar para otro lado). Es lo que hace la mayoría.

(5) ¿Qué es el sistema de archivos virtual? Describa brevemente sus dos funcionalidades más importantes.

El sistema de archivos virtual es una capa de abstracción de la implementación de los diferentes sistemas de archivos. Provee dos funcionalidades importantes:

- Propone una interfaz genérica de sistema de archivo que es independiente del tipo de sistema de archivo. De esta forma, se logra un acceso transparente al sistema de archivos.
- Propone un bloque de control de archivo virtual que puede representar tanto archivos locales como remotos.

(6) Describa brevemente la estructura RAID 5 e indique los pasos requeridos para realizar la escritura de 1 bloque en RAID 5.

Un sistema con RAID 5 se caracteriza por:

- Usar división a nivel de bloques.
- Los datos de paridad son distribuidos entre todos los discos del RAID.
- Soporta la falla de hasta un disco.

Pasos para realizar la escritura de 1 bloque:

- Leer el bloque de datos antiguo.
- Leer el bloque de paridad antiguo.
- Comparar el bloque de datos antiguo con la solicitud de escritura. Por cada bit que se ha invertido (de 0 a 1 o de 1 a 0) en el bloque de datos, dar vuelta el bit correspondiente en el bloque de paridad.
- Escribir el bloque de datos nuevo.
- Escribir el bloque de paridad nuevo.

(7) Describa brevemente qué es la Hiperpaginación. ¿Qué efectos puede tener sobre el sistema operativo?

Cuando un proceso pasa más tiempo paginando que ejecutando se dice que sufre de hiperpaginación. La hiperpaginación degrada significativamente el rendimiento del sistema.

(8) En un sistema de paginación bajo demanda:

i. Explique brevemente un esquema simple para traducir una dirección virtual a una física.

ii. Enumere los elementos de hardware necesarios para su funcionamiento y la función que cumple cada uno.

i. Las direcciones virtuales se componen de un número de página y un desplazamiento (offset). El número de página es el índice sobre una tabla de páginas y el desplazamiento es la referencia dentro de la página. La tabla de páginas contiene el mapeo entre cada página y el marco en memoria en el que se encuentra.

ii.

- Memory Management Unit (MMU): es la encargada de realizar la traducción de direcciones virtuales a físicas.
- Translation Look-aside Buffer (TLB): mantiene en caché la traducción de direcciones realizada por la MMU. Cada entrada consiste de dos partes: una clave y un valor (el número de frame). El cache TLB debe ser asociativa y de rápido acceso para que sea efectiva.
- Registro PTBR: mantiene una referencia a la tabla de página en memoria principal y es utilizado por la MMU para realizar la traducción.

(9) Explique brevemente cómo funciona:

i. una E/S sincrónica.

ii. una E/S asincrónica.

- i. Los procesos que realizan pedidos de E/S se bloquean hasta que el pedido finalice. La operación se realiza de manera secuencial.
- ii. El proceso que realiza el pedido de E/S continúa su ejecución y la operación es ejecutada en paralelo. Cuando finaliza la E/S, el sistema operativo le notifica al proceso (p.ej. usando una señal).

(10) Describa brevemente qué es:

i. Discretionary Access Control

ii. Mandatory Access Control

- i. Los usuarios son dueños de objetos del sistema. Cada usuario controla el acceso a los objetos que es dueño y puede permitir o denegar el acceso de otros usuarios.
- ii. El acceso a los objetos es controlado por una política global del sistema. Ningún usuario puede alterar ese acceso.

(11) Describa brevemente qué es un Hipervisor de Tipo 1.

Un hipervisor de tipo 1 ejecuta en modo kernel. Cada VM ejecuta como un proceso de usuario en modo usuario con modo kernel virtual y modo usuario virtual. El hipervisor utiliza la técnica de trap-and-emulate para ejecutar las instrucciones sensibles a nombre del sistema operativo virtualizado en cada VM. Cuando una VM ejecuta una instrucción sensible se produce una trap que es atrapada y procesada por el hipervisor.