

Segundo Parcial 2017-2018

1. El bit de referencia de la tabla de páginas indica si la página está o no en memoria.
2. La asignación variable de marcos tiene en cuenta el conjunto de trabajo para variar o no la asignación de marcos.
3. La gestión de memoria virtual es transparente al programador.
4. Todo sistema de memoria virtual necesita utilizar una tabla de páginas.
5. Las TLBs contienen las últimas páginas referenciadas del proceso.
6. En un esquema de memoria con paginación, cuyo marco de página sea de 8K, son necesarios 10 bits para referenciar el desplazamiento.
7. En un esquema de memoria virtual con paginación, cada entrada de la tabla de páginas tiene un bit para indicar si la página está o no cargada en memoria.
8. La variación del número de marcos en función de la tasa de fallos de página es un mecanismo para decidir qué página debe ser expulsada.
9. En un sistema que utiliza memoria virtual no es necesario cargar una instrucción en memoria principal para ejecutarla.
10. El algoritmo de reemplazo LRU introduce más sobrecarga que el algoritmo del reloj.
11. En un esquema de memoria virtual con paginación, cada entrada de la tabla de páginas tiene un bit para indicar si la página ha sido o no modificada.
12. La traducción de direcciones es más rápida en paginación que en segmentación.
13. La gestión de memoria virtual resulta eficiente gracias a que se cumple el Principio de Localidad de Referencia.
14. Las tablas multinivel reducen el espacio en memoria dedicado al almacén de la tabla de páginas.
15. Las TLBs reducen en general el tiempo requerido para traducir direcciones virtuales.
16. En el escenario 1, Utilizando el algoritmo óptimo, para cargar la página 3 se reemplazaría la 5.
17. En el escenario 1, utilizando LRU para cargar la página 3 se reemplaza la 10.
18. En el escenario 1, Usando el algoritmo del reloj, la segunda vez que se referencia a la página 10 se sustituye la 1.
19. Para conocer si una E/S ha terminado, utilizando E/S programada, el SO lee repetidamente el registro de estado del dispositivo.
20. Las peticiones de E/S se encolan en colas de IORBs.
21. El IORB es una estructura de datos que crea el gestor de E/S y guarda la información de una petición de e/s.
22. La gestión de e/s facilita la comunicación entre procesos que estén en la misma máquina.
23. Cada petición de e/s genera un hilo en el gestor dependiente del dispositivo.

24. El controlador hardware recibe instrucciones del gestor independiente del dispositivo.
25. Si se utiliza DMA la transferencia de información entre memoria y el dispositivo la realiza el controlador hardware.
26. El driver de un dispositivo que debemos descargar cuando usamos un dispositivo nuevo constituye el manejador independiente del dispositivo.
27. El algoritmo Scan puede producir inanición de peticiones.
28. El tiempo de latencia es el tiempo que tarda la cabeza de lectura en colocarse sobre la pista en la que se sirve la petición.
29. El tiempo de Búsqueda depende de la velocidad de rotación del disco.
30. El algoritmo SSTF puede producir inanición de peticiones.
31. El tiempo de transferencia depende de la velocidad de rotación del disco.
32. El algoritmo C-Look no discrimina pistas de los extremos frente a pistas centrales.
33. En el escenario 2, si se utiliza el algoritmo LOOK la siguiente petición en ser atendida será la 40.
34. En el escenario 2, Utilizando SSTF la tercera petición en atenderse será la 40.
35. En el escenario 2, Utilizando SCAN, la última pista en visitarse en este barrido será la 125.
36. La técnica del journaling permite accesos más rápidos a los ficheros.
37. En NTFS, la MFT puede almacenar parte del propio fichero.
38. Los directorios son ficheros con formato específico conocido por el sistema operativo.
39. El gestor de ficheros se comunica con el gestor de dispositivos.
40. En un sistema de ficheros FAT, para localizar el bloque lógico 0 de un fichero es necesario el acceso a una entrada de la FAT.
41. Entre la información de un fichero, el sistema guarda información para su localización en disco.
42. Los sistemas de ficheros más modernos, como los derivados de ext3 o NTFS utilizan un árbol equilibrado para la localización de los bloques de un fichero en disco.
43. Un fichero es una unidad de almacenamiento lógico no volátil que agrupa un conjunto de información relacionada entre sí bajo un nombre.
44. El almacenamiento contiguo de datos ya no se usa en ningún tipo de dispositivo por ser muy poco flexible.
45. En el escenario 3, En la entrada 10 de la FAT se guarda el número 48.
46. En el escenario 3, En la entrada 0 de la FAT se guarda el número 10.
47. En el escenario 3, Para acceder al bloque 132 el sistema tiene que leer primero los bloques 10, 48 y 256.
48. En el escenario 4, En la entrada 0 de la tabla de índices se guarda el número 10.
49. En el escenario 4, En la entrada 10 de la tabla de índices se guarda el número del bloque que contiene los números de bloque 69, 25 y 56.
50. En el escenario 4, En la entrada 10 de la tabla de índices se guarda el número 48.

Escenario 1:

Sea un sistema de gestión de memoria con paginación, con 4 marcos de página asignados a un proceso. Dicho proceso realiza la siguiente secuencia de acceso a páginas:

10,5,1,6,3,5,10,3,1,5,3,1

Escenario 2:

Sea un disco duro con 4 cabezas, 128 pistas por cara y 64 sectores por pista. Un bloque está compuesto por 4 sectores. En un momento dado, cuando la cabeza de lectura/escritura se encuentra sobre la pista 30 (procedente de la 20), la lista de peticiones pendientes (ordenadas por orden de llegada) son sobre los siguientes cilindros:

27, 35, 1, 90, 125, 18, 22, 40

Escenario 3:

Sea un sistema de ficheros FAT. Sea un fichero con los siguientes bloques físicos asignados:

10, 48, 256, 132, 24, 84, 79, 93, 253, 30, 69, 25, 56

Escenario 4:

Sea un sistema de ficheros con tabla de índices de 13 entradas. Sea un fichero con los siguientes bloques físicos asignados:

10, 48, 256, 132, 24, 84, 79, 93, 253, 30, 69, 25, 56