

Segundo Parcial 2014-2015

1. Los sistemas actuales realizan la traducción de direcciones lógicas a físicas en tiempo de carga.
2. La protección de la memoria impide el acceso a direcciones de memoria fuera del espacio de direcciones del proceso que solicita el acceso
3. La protección, al igual que la traducción, la lleva a cabo la Unidad de Gestión de Memoria en los procesadores actuales de uso común
4. Un esquema de asignación de memoria real puede provocar fallos de página.
5. En un esquema de asignación contigua de memoria con particiones fijas, el tamaño de las particiones depende del tamaño de los procesos que sean cargados.
6. Dado un sistema de gestión de memoria virtual con paginación, en el que se emplea tabla de páginas de un único nivel, con páginas de 4K. Dado un proceso cuyo espacio de direcciones ocupa 47982 bytes, este proceso necesitará una tabla de páginas de 12 entradas.
7. En un sistema de gestión de memoria virtual con paginación, la tabla de páginas de un proceso almacena información que permite detectar los fallos de páginas de ese proceso
8. En un esquema de memoria virtual con paginación que disponga de TLB, algunas de las páginas del proceso estarán cargadas en la TLB
9. Dado un sistema de gestión de memoria virtual con paginación, en el que se emplea tabla de páginas de dos niveles, en los que cada entrada de la tabla ocupa 2 bytes, y tamaño de página de 2K. Dado un proceso de tamaño 1Gbyte de los que se usan únicamente los 4.700.000 bytes de las posiciones más bajas (del byte 0 al 4.699.999). Será necesarios 1026 Kbytes para almacenar la tabla de páginas.
10. En un esquema de gestión de memoria virtual con segmentación, el sistema integra el concepto de región (código, datos, pila...) que puede compartir, proteger, etc. En un sistema de paginación no hay este concepto.
11. Un esquema de asignación de memoria virtual mejora el grado de multiprogramación con respecto a un esquema de asignación de memoria física
12. La paginación de memoria es apropiada para gestionar regiones.
13. La memoria virtual se basa en el principio de localidad.
14. En un sistema de memoria virtual con bus de 16 bits el espacio de direcciones lógicas puede abarcar 216 direcciones.
15. Para traducir una dirección virtual a física, lo primero es mirar en la tabla de páginas. Si eso falla, se consulta el TLB.
16. En gestión de memoria virtual, FIFO es una política válida de lectura.
17. En gestión de memoria virtual, una política LRU puede verse afectada por la anomalía de Belady.

18. En un sistema de memoria virtual basado en el algoritmo del reloj, con una memoria física asignada de 4 marcos, y cuyos contenidos son: (Página=12|bit de referencia=1) (15|1) (18|0) (45|1), y el "puntero al siguiente marco" apunta al segundo, cuando se solicita la página 49, el segundo marco se usa para cargar esa página 49, y la 15 abandona la memoria física.
19. En un sistema de memoria virtual basado en el algoritmo del reloj, con una memoria física asignada de 4 marcos, y cuyos contenidos son: (Página=12|bit de referencia=1) (15|1) (18|0) (45|1), y el "puntero al siguiente marco" apunta al segundo, cuando se solicita la página 49, el tercer marco se usa para cargar esa página 49, y la 18 abandona la memoria física.
20. En un sistema de memoria virtual basado en el algoritmo del reloj, con una memoria física asignada de 4 marcos, y cuyos contenidos son: (Página=12|bit de referencia=1) (15|1) (18|0) (45|1), y el "puntero al siguiente marco" apunta al segundo, cuando se solicita la página 49, después de terminar de procesar esa petición el bit de referencia para el segundo marco es 0.
21. En un sistema de memoria virtual basado en el algoritmo del reloj, con una memoria física asignada de 4 marcos, y cuyos contenidos son: (Página=12|bit de referencia=1) (15|1) (18|0) (45|1), y el "puntero al siguiente marco" apunta al segundo, cuando se solicita la página 49, después de terminar de procesar esa petición el "puntero al siguiente marco" apunta al tercero.
22. El controlador hardware del dispositivo recibe instrucciones del controlador software (driver) del sistema operativo
23. La comunicación con el controlador hardware a través de puertos utiliza instrucciones máquina privilegiadas
24. Cualquier programa de usuario puede comunicarse directamente con el controlador hardware a través de las operaciones port-in y port-out
25. El sistema operativo deja la operación a realizar por el dispositivo en los registros de control del controlador hardware
26. El modo de transferencia con Acceso Directo a Memoria permite al controlador hardware el acceso a la memoria principal del ordenador
27. Una transferencia con Acceso Directo a Memoria no es una entrada/salida por interrupciones
28. Cada petición que recibe el gestor de dispositivos genera un hilo de ejecución en el gestor independiente del dispositivo
29. Cada petición que recibe el gestor de dispositivos genera un hilo de ejecución en el gestor dependiente del dispositivo
30. Cuando un dispositivo finaliza una operación genera una interrupción que será trasladadas al manejador independiente del dispositivo
31. La cola de IORBs es utilizada tanto por el manejador dependiente como por el independiente del dispositivo.
32. Si un disco duro tiene 256 sectores por cilindro, y 512 bytes por sector, 4 cabezas y una capacidad total de 256 MB podemos concluir que tiene 2048 pistas por cada cara.
33. En gestión de E/S, el software dependiente del dispositivo recibe IORBs.

34. En un disco, el número de caras es igual que el número de cabezas de lectura/escritura.
35. El tiempo de latencia depende de la velocidad de rotación.
36. En políticas de planificación de disco, FCFS es muy fácil de implementar, y por tanto ofrece un rendimiento excelente.
37. En políticas de planificación de disco, las que pueden causar inanición son FCFS y SSTF.
38. En políticas de planificación de disco, las únicas que tratan todas las peticiones igualmente (es decir, no favorecen pistas centrales o extremas) son Scan y Look.
39. Cuando se solicita un bloque, si está en la caché de bloques, esa petición no llega a procesarse en los manejadores de dispositivo (ni dependientes ni independientes).
40. El tiempo de búsqueda (en un disco) depende de la velocidad de rotación.
41. Un disco usa la política de planificación Look. La cabeza está inicialmente en el cilindro 0. Las peticiones pendientes son 10, 46, 90, 24, 57, 32. Cuando está procesando la 32, llega una petición para 14. Cuando está procesando 46, llega una petición para 59. El orden de servicio de las peticiones es: 10 24 32 46 57 59 90 14.
42. Un disco usa la política de planificación Look. La cabeza está inicialmente en el cilindro 0. Las peticiones pendientes son 10, 46, 90, 24, 57, 32. Cuando está procesando la 32, llega una petición para 14. Cuando está procesando 46, llega una petición para 59. El orden de servicio de las peticiones es: 10 24 32 46 57 90 59 14.
43. Los Sistemas de ficheros que usan habitualmente Windows y Linux interpretan los ficheros del usuario como una secuencia de bytes
44. Los ficheros ejecutables, o los ficheros de texto son ejemplos de ficheros cuyo formato es conocido para el sistema operativo
45. Actualmente no se usa el almacenamiento contiguo de ficheros
46. Sea un sistema de ficheros tipo FAT32 con bloques de 512 bytes, y un fichero que tiene asignados los siguientes bloques de disco: 1018, 456, 8358, 143, 294,2984. Para localizar el byte 3000 el sistema deberá acceder 5 veces a la FAT
47. Sea un sistema de ficheros tipo ext2, con asignación indexada con tablas de índices de 13 entradas, con bloques de 512 bytes, y un fichero que tiene asignados los siguientes bloques de disco: 1018, 456, 8358, 143, 294,2984. Para localizar el byte 3000 el sistema será más rápido que si fuese FAT32.
48. La técnica del Journaling permite acelerar la velocidad de acceso a los datos de los ficheros.
49. NTFS usa tablas de índices para la localización de los bloques de los ficheros
50. En la tabla MFT que usa NTFS cada fichero tiene una entrada que constituye el nodo raíz del árbol equilibrado
51. Las agrupaciones son conjuntos de ficheros del mismo directorio
52. El tiempo que tarda el sistema en localizar un bloque de un fichero en un sistema de asignación contigua depende de la posición de éste en el fichero (cuanto más lejos del principio más se tarda en localizar)

53. Un bloque es lo mismo que un sector.
54. En el sistema de ficheros ISO9660, el contenido de un fichero es contiguo.
55. En el sistema de ficheros de Unix System V, el número total de i-nodos disponibles se ajusta dinámicamente dependiendo de las necesidades.
56. En NTFS, un fichero (incluso uno que no esté vacío) podría no necesitar ningún bloque de datos.
57. En el sistema de gestión de ficheros de un sistema operativo, el sistema de ficheros virtual pide bloques directamente al manejador independiente del dispositivo.
58. Una configuración RAID 5 distribuye la información de paridad entre todos sus discos
59. El journaling y los mapas de bits son dos técnicas para hacer los sistemas de ficheros más seguros.
60. Supongamos un sistema de ficheros con asignación indexada por tablas de índices (estilo System V). Cada tabla, sea del nivel que sea, tiene 13 entradas, y la primera tiene 10 referencias directas, 1 indirecta, 1 con doble indirección y 1 con triple indirección. Un fichero tiene 11 bloques: 54, 22, 45, 34, 35, 36, 129, 86, 470, 212, 46. La onceava entrada de la primera tabla de índices contiene 46
61. Supongamos un sistema de ficheros con asignación indexada por tablas de índices (estilo System V). Cada tabla, sea del nivel que sea, tiene 13 entradas, y la primera tiene 10 referencias directas, 1 indirecta, 1 con doble indirección y 1 con triple indirección. Un fichero tiene 28 bloques. Necesitará 3 tablas de índices en total.
62. Supongamos un sistema de ficheros con asignación indexada por tablas de índices (estilo System V). Cada tabla, sea del nivel que sea, tiene 13 entradas, y la primera tiene 10 referencias directas, 1 indirecta, 1 con doble indirección y 1 con triple indirección. Un fichero tiene 28 bloques. Necesitará 4 tablas de índices en total.