

(Febrero 2007 M1 y M2)

1. En un sistema de ficheros de inodos de 32 bits, con un tamaño de bloque de 4KB, ¿cuál es el tamaño máximo en bytes de un fichero?

- a) $10 + 256 + 256^2 + 256^3$
- b) $(10 + 256 + 256^2 + 256^3) * 4096$
- c) $10 + 1024 + 1024^2 + 1024^3$
- **d) $(10 + 1024 + 1024^2 + 1024^3) * 4096$**
- $32 \text{ bits} = 4 \text{ bytes}$
- $4096 \text{ bytes bloque} / 4 \text{ bytes puntero} = 1024 \text{ punteros en un bloque}$

(Enero 2009, Febrero 2008)

6. En la práctica, con tamaño de bloque igual a 1K y punteros de 32 bits, ¿cuál es el tamaño máximo aproximado de los ficheros?

- **a) 16 GB $8 + 1024 + 1024^2 + 1024^3$**
- $32 \text{ bits} = 4 \text{ bytes}$
- $1024 \text{ bytes bloque} / 4 \text{ bytes puntero} = 256 \text{ punteros en un bloque}$
- $8 + 256 + 256^2 + 256^3 = 16 \text{ millones de bloques} * 1024 \text{ bytes bloque}$
- = 16 GB
- b) 64 MB
- c) 4 GB
- d) 16 MB

(Febrero 2006 M1 y M2)

Si se implementa un sistema de ficheros de i-nodos estándar de 32 bits y tamaño de bloque de 4 KB. ¿Cuántos bloques de datos en total puede tener cada fichero?

- a) $10 + 256 + 256^2 + 256^3$
- **b) $10 + 1024 + 1024^2 + 1024^3$** Fuente: A1-7, L-577
- $4\text{KB} = 4096\text{B}; 32\text{b} = 4\text{B}; 4096/4 = 1024$
- c) $10 + 3 * 256$
- d) $10 + 3 * 1024$

(Septiembre 2005 M1 y M2)

4. Si se implementa un sistema de ficheros con i-nodos, donde el i-nodo principal tiene diez punteros directos y tres indirectos. Si los indirectos apuntan a su vez a otros bloques índices secundarios con 256 punteros. ¿Cuántos bloques de datos en total puede tener cada fichero?.

- **a) $10 + 256 + 256^2 + 256^3$** Fuente: A1-7, L-577
- b) $13 + 256 + 256^2 + 256^3$
- c) $10 + 256 + 256^2$
- d) $10 + 256^3$

(Febrero 2005)

20. Si se implementa un sistema de ficheros con i-nodos, donde el i-nodo principal tiene nueve punteros directos y tres indirectos. Si los indirectos apuntan a su vez a otros bloques

índices secundarios con 256 punteros en vez de los 12. ¿Cuántos bloques de datos en total puede tener cada fichero?.

- **a) $9+256+256^2+256^3$** Fuente: A1-7, L-577
- b) $12+256+256^2+256^3$
- c) $12+256^3$
- d) $9+256^3$

(Septiembre 2004)

15. Si se implementa un sistema de ficheros con i-nodos, donde el i-nodo principal tiene diez punteros directos y tres indirectos. Si los indirectos apuntan a su vez a otros bloques índices secundarios con 256 punteros en vez de los 13. ¿Cuántos bloques de datos en total puede tener cada fichero?

- **a) $10+256+256^2+256^3$** Fuente: A1-7, L-577
- b) $13+256+256^2+256^3$
- c) $10+256+256^2$
- d) $10+256^3$

(Septiembre 2002)

20. Si se implementa un sistema de ficheros con i-nodos, donde el i-nodo principal tiene 9 punteros directos y 3 indirectos. Si los indirectos apuntan a su vez a otros i-nodos secundarios del mismo tamaño (similar a la práctica), pero con 24 punteros en vez de los 12. ¿Cuántos i-nodos puede tener un fichero como máximo? (total, contando el principal).

- a) 600
- **b) Ninguna de las otras opciones.** Fuente: A3-8
- $(1) \text{ directo} + (1) \text{ 1er nivel} + (1 + 24) \text{ 2o nivel} + (1 + 24 + 24 \cdot 24) \text{ 3er nivel}$
= 628
- c) 424
- d) 604

(Febrero 2002)

18. Si se implementa un sistema de ficheros con i-nodos, donde el i-nodo principal tiene los 10 punteros directos y 3 indirectos. Si los indirectos apuntan a su vez a otros i-nodos secundarios del mismo tamaño (similar a la práctica), pero con 20 punteros en vez de los 13. ¿Cuántos i-nodos puede tener un fichero como máximo? (total, contando el principal).

- **a) 444** Fuente: A3-8
- 1 principal
- 1 inodo indirecto de primer nivel
- 1 inodo indirecto de primer nivel y los 20 a los que apunta (inodos de segundo nivel)
- 1 inodo indirecto de primer nivel, los 20 a los que apunta, y los 20 a los que apuntan cada uno de estos (20.20)
- $(1) + (1) + (1 + 20) + (1 + 20 + 20 \cdot 20) = 400 + 40 + 4 = 444$
- b) 4
- c) 424

- d) 44

(Febrero 2002)

19. ¿Cuántos bloques puede tener el fichero de la pregunta anterior?

- a) 400
- b) 70
- c) 8440
- **d) 8430** Fuente: A3-8
- $10 + 20 + 20 \cdot 20 + 20 \cdot 20 \cdot 20 = 8000 + 400 + 30 = 8430$

(Febrero 2008)

12. Si en la práctica (9 punteros directos, 3 indirectos y soporte de “ficheros malos”, con bloques de 1024 bytes) se escriben sólo unos bytes en el bloque lógico 300, ¿cuántos bloques de la zona de “datos” se usarán?

- a) 1
- b) 2
- **c) 3**
- Explicación: El bloque lógico 300 es un indirecto de segundo nivel ($9 + 256 < 300 < 9 + 256 + 256^2$), entonces ocupa un bloque de datos para los datos del fichero, más dos bloques adicionales para los índices indirectos, o sea la c)
- d) No se puede calcular.

(Febrero 2008)

14. En el mismo sistema anterior (9 punteros directos, 3 indirectos y soporte de “ficheros malos”, con bloques de 1024 bytes) ¿cuántas lecturas de bloques -total, el inodo ya está en memoria- se necesitan para leer el byte en la posición 300.000?

- a) 1
- b) 2
- **c) 3**
- $300.000 / 1024 \text{ bytes bloque} = 292 \text{ bloque}$
- $9 + 256 + 256^2 > 292 > 9 + 256$
- d) 4

(Febrero 2007 M1 y M2)

20. En un sistema de ficheros basados en inodos, con 10 punteros directos el tamaño mínimo de los ficheros es superior a los 30 KB. ¿Qué tamaño de bloque elegiría?

- a) 512 Bytes
- b) 1 KB
- c) 2 KB
- **d) 4 KB**
- Porque así los ficheros pequeños utilizarán sólo punteros directos ($30/4 = 7,5$). Si fuese 2KB un fichero de 30KB utilizaría los indirectos también ($30/2 = 15$). ¿?

(Enero 2009)

7. En el mismo sistema anterior (bloque 1K, punteros 32 bits, práctica), ¿cuántas lectoras de bloques -total, el inodo ya está en memoria-, se necesitarán para leer el bloque lógico 500 de un fichero?

- a) 1
- b) 2
- **c) 3**
- $8 + 256 + 256^2 > 500 > 8 + 256 = 2^\circ$ nivel
- Lee el bloque apuntado por el inodo indirecto de 2º nivel, éste apunta a otro bloque de punteros, y finalmente el bloque de datos
- d) 4

(Febrero 2006 M1 y M2)

2. ¿Dónde se almacena la información número de enlaces de directorios?

- a) En la entrada del fichero en el directorio.
- b) Se necesita un área especial en el sistema de ficheros.
- **c) En el inodo del fichero.** Fuente: A1-8
- d) En la raíz de directorios.

(Enero 2003)

7. Si hubiese que extender la práctica para que soporte el enlace de ficheros, es decir que distintas entradas del directorio hagan referencia el mismo fichero de datos, ¿que estructura de datos hay que cambiar?:

- a) Hay que cambiar la estructura de los inodos y el directorio.
- **b) Sólo hay que agregar un campo a los inodos.** Fuente: A3-9
- c) Sólo hay que cambiar la estructura del directorio.
- d) No se puede implementar a menos que se cambie radicalmente la mayoría de las estructuras.

(Enero 2009, Septiembre 2007, Septiembre 2006, Febrero 2006 M1 y M2, Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Septiembre 2003, Enero 2003)

3. En la secuencia de finalización de procesos generada en el algoritmo del banquero:

- **a) Está en un estado seguro si todos los procesos aparecen en la lista.**
- Fuente: A2-12, A3-4, Examen
- b) Está en un estado seguro si ningún proceso queda en la lista.
 - c) La secuencia indica el orden en que deben acabar los procesos.
 - d) Está en un estado inseguro si queda algún proceso en la lista.
 - b) Está en un estado inseguro si todos los procesos aparecen en la lista.
 - d) Si no están todos los procesos, seguro que se produce un interbloqueo.
-
- a) La secuencia de indica el orden en que deben acabar los procesos F
 - **b) Si no están todos los procesos se debe rechazar la asignación**
 - c) Si están todos los procesos se debe rechazar la asignación F

- d) Está en un estado inseguro si todos los procesos aparecen en la lista F

(Enero 2009, Febrero 2007 M1 y M2)

1. El algoritmo del banquero

- a) **Es para prevenir que se produzcan deadlocks**
- b) Detecta dinámicamente los deadlocks
- c) Soluciona y recupera al sistema de situaciones de deadlock
- d) Hace que no se cumplan una o más de las cuatro condiciones necesarias
- x) Asegura exclusión mutua con múltiples procesos

(Enero 2009)

20. ¿Cuál es la dificultad fundamental para el uso generalizado del algoritmo del banquero?

- a) La complejidad del algoritmos
- b) La imposibilidad de tratar con clases de múltiples instancias
- c) **La dificultad de conocer de antemano las necesidades de recursos de cada proceso**
- d) La imposibilidad de tratar con un número variable de procesos activos

(Enero 2003)

14. Durante la ejecución del algoritmo del estado seguro no se encuentra ningún vector de la matriz “Necesidad” que sea menor o igual a los recursos disponibles en el vector “Trabajo”:

- a) Significa que hay un interbloqueo.
- b) **Significa que el estado es inseguro.** Fuente: A2
- ¿En el caso de que no hayan sido analizados todos ya?
- c) Significa que el estado es seguro.
- d) Significa que los procesos no pueden acabar.

(Septiembre 2007, Febrero 2006 M1 y M2, Septiembre 2005 M1 y M2, Septiembre 2004, Enero 2004 M2 y M1, Enero 2003, Septiembre 2002, Febrero 2002)

4. ¿Que técnica es la más comúnmente utilizada para optimizar la búsqueda de dispositivo/ bloque en los sistemas de buffer-cache?

- a) Arrays de punteros dinámicos.
- b) Arrays de punteros estáticos.
- c) Productor/consumidor.
- d) **Tablas de dispersión (hashing).** Fuente: L-525, A1-10, A3-2, A0-11
- a) Árboles balanceados F
- b) **Tablas de dispersión**
- c) Arrays de punteros dinámicos
- d) Arrays de punteros estáticos

(Febrero 2007 M1 y M2, Febrero 2005)

9. ¿Que técnica usaría para optimizar la búsqueda de dispositivo/bloque en un sistema de buffer-cache?

- a) Arrays de punteros dinámicos.
- b) Tablas de dispersión (hashing), con el tamaño de la tabla igual a un número primo.
- c) Productor/consumidor.
- **d) Tablas de dispersión (hashing), con el tamaño de tabla potencia de 2 y funciones basadas en el número áureo.** Fuente: A1-10, ¿número primo es el rango no el tamaño?

(Septiembre 2008)

12. ¿Qué técnica es la más comúnmente utilizada para optimizar la búsqueda en los directorios de un sistema de ficheros basado en inodos? (Pista: Los directorios son de tamaño variable a interesa también acceder por orden alfabético.)

- a) Árboles balanceados.
- **b) Tablas de dispersión.** Fuente: A0-9
- c) Arrays de punteros dinámicos.
- d) Arrays de punteros estáticos.

(Febrero 2006 M1 y M2, Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Enero 2003)

5. ¿Cuál de los siguientes métodos de gestión de memoria es más apto para minimizar la fragmentación en sistemas con muchos procesos activos?

- a) Particiones fijas.
- b) Particiones variables.
- **c) Paginación.** Fuente: L-312, A3-4
- d) Segmentación.

(Febrero 2005)

1. ¿Cuál de los siguientes sistemas de gestión de memoria tiene la menor fragmentación interna media por proceso?

- a) Particiones fijas.
- b) Particiones variables.
- **c) Segmentación. ¿?**
- d) Paginación. ¿?
- ¿Razonamiento: Partición fija tiene más fragmentación interna, y las otras dos no tienen, esta -paginación- es la que tiene y su media es la menor?

(Septiembre 2004, Febrero 2002)

16. ¿Cuál de los siguientes sistemas de gestión de memoria sufre de menos fragmentación interna?

- a) Particiones fijas.
- b) Particiones variables.
- **c) Segmentación. ¿?** Fuente: L-312, A3-4
- d) Paginación.

(Enero 2009, Febrero 2008)

13. ¿Cuál es el problema fundamental del SSTF para planificación de E/S a disco?

- **a) Puede provocar tiempos de latencias muy elevados para algunas operaciones**
 - b) Favorece a los que están ubicados en las pistas de los extremos
 - c) Que da demasiada prioridad al posicionamiento de las cabezas
 - d) Es muy costoso de implementar
-
- **a) Favorece a los ficheros ubicados en las pistas centrales**
 - b) Favorece a los que están ubicados en las pistas de los extremos
 - c) Se pierde mucho tiempo en el posicionamiento de las cabezas
 - d) Favorece a los ficheros ubicados en las pistas centrales

(Febrero 2007 M1 y M2)

4. ¿Cuál de las siguientes técnicas de planificación de E/S a disco se aproxima más al round-robin?

- a) SSTF
- b) SCAN
- c) CSCAN
- **d) CFQ (complete fair queue)** Fuente: A0-16, [blog de Galli](#)

(Febrero 2008)

3. ¿Cómo se pueden reducir las enormes latencias, "esperas infinitas", para algunas operaciones de E/S a disco?

- a) Con el SCAN
- b) Con el SSTF
- **c) Con colas múltiples**
 - Explicación: Las colas múltiples fueron específicamente desarrolladas para limitar los tiempos de latencias que sufrían las anteriores, fundamentalmente las variantes de SCAN y SSTF. Por otro lado, el CFQ (complete fair queue) sí es una técnica que también permite limitar el tiempo de espera máximo, pero es mucha más moderna. De todas formas no estaba entre las cuatro técnicas que se planteaban.
- d) Con el FIFO

(Febrero 2007 M1 y M2, Febrero 2005)

15. ¿Cuál de los siguientes algoritmos de planificación de disco es comúnmente llamado el ascensor?

- **a) SCAN.** Fuente: A0-16, A1-16, A3-4, L-510, siguiente pregunta
- b) SSTF (primero el más corto).
- c) PEPS (primero en entrar, primero en salir).
- d) UEPS (último en entrar, primero en salir).

(Enero 2004 M2 y M1)

15. El algoritmo del ascensor o SCAN sirve para:

- a) Planificar procesador.
- b) Seleccionar las páginas candidatas a ser reemplazadas.
- **c) Planificación operaciones de E/S a disco.** Fuente: L-510, A3-4, A1-16
- d) Para estimar el conjunto de trabajo de los procesos.

(Septiembre 2004)

6. Indicar cual de los siguientes algoritmos de planificación de disco funciona mejor cuando son pocos los procesos que requieren acceso a disco.

- a) SCAN.
- b) SSTF (primero el más corto).
- **c) PEPS / FIFO (primero en entrar, primero en salir).** Fuente: L-507
- d) UEPS / LIFO (último en entrar, primero en salir).

(Septiembre 2004)

6. Indicar cual de los siguientes algoritmos de planificación de disco es el más usado habitualmente en sistemas de tiempo compartido.

- **a) SCAN.** Fuente: L-507, A1-16, A3-4
- b) SSTF (primero el más corto).
- c) PEPS / FIFO (primero en entrar, primero en salir).
- d) UEPS / LIFO (último en entrar, primero en salir).

(Septiembre 2006, Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Septiembre 2003)

16. Para la planificación de disco en un sistema simple y de mono-programación/donde sólo un proceso accede a ficheros:

- **a) El FIFO es el más/único/mejor razonable.** Fuente: L-507, A3-6, Internet
- b) El primer parámetro a minimizar es el tiempo de latencia de giro.
- c) Lo importante es maximizar la transferencia de datos para cada proceso.
- d) Se debería usar un SCAN o similar.
- x) El primer parámetro es minimizar es el tiempo de posicionamiento de las cabezas.

(Enero 2003)

18. Para la planificación de disco en un sistema de multiprogramación:

- a) El mejor algoritmo es el FIFO.
- b) El primer parámetro a minimizar es el tiempo de latencia de giro.
- c) Lo importante es maximizar la transferencia de datos para cada proceso.

- **d) El primer parámetro es minimizar es el tiempo de posicionamiento de las cabezas.¿?**

(Febrero 2007 M2)

16. ¿Cuál de los siguientes parámetros es el que más afecta al rendimiento de los dispositivos de bloque?

- **a) Tiempo de posicionamiento** Fuente: A1-12
- Tiempo de posicionamiento > Latencia de giro > Tiempo de transferencia
- b) Latencia de giro
- c) Tiempo de transferencia
- d) Fragmentación interna de bloques

(Septiembre 2007, Febrero 2006 M1 y M2)

6. Para la planificación de disco en un sistema de tiempo compartido:

- a) El FIFO es el más razonable. F
- b) El primer parámetro a minimizar es el tiempo de latencia de giro.
- **c) Debería intentar minimizar el tiempo de búsqueda y controlar las latencias.**
- d) Se debería usar un LIFO o similar para disminuir las latencias.
- a) El mejor algoritmo es el FIFO F
- b) El primer parámetro a minimizar es el tiempo de latencia de giro
- c) Lo importante es maximizar la transferencia de datos para cada proceso
- **d) El primer parámetro a minimizar es el tiempo de posicionamiento de las cabezas**

(Febrero 2006 M1 y M2, Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Enero 2003)

7. Un proceso ha generado un “fallo de página”:

- a) Significa que hubo un error de ejecución.
- **b) La página referenciada no está en memoria RAM.** Fuente: A1-20, L-349, A4-10
- c) Una página de ese proceso a sido seleccionada como víctima para ser intercalada a disco.
- d) Significa que la página no ha sido encontrada en el TLB.
- **x) Posiblemente se esté usando paginación por demanda** Fuente: L-360

(Enero 2009)

16. Las estructuras de las tablas de páginas:

- **a) Están definidas y limitadas por el diseño del procesador**
- b) Sólo están limitadas por el diseño del software del SO

- c) Dependen del del tiempo y cantidad de memoria RAM
- d) Dependen de los algoritmos y niveles de seguridad que se deseen implementar en el sistema

(Febrero 2006 M1 y M2, Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Enero 2003, Febrero 2002)

8. El tamaño de las tablas de páginas de cada proceso:

- a) Es proporcional al tamaño de página. F
- **b) Es inversamente proporcional al tamaño de página.** Fuente: A3-8, L-345-347, [Interné](#)
- c) Es proporcional al tamaño del TLB. F
- d) Es proporcional a la cantidad de procesos activos.
- c) Es proporcional al tamaño de la memoria intercalada a disco. F

(Febrero 2002)

8. El tamaño de las tablas de páginas:

- a) Es proporcional al tamaño de las páginas. F
- **b) Es (inversamente) proporcional al tamaño de las páginas y proporcional a la cantidad de memoria virtual usada por los procesos.** Fuente: A3-8, [Interné](#), [Interné2](#)
- c) Es proporcional al tamaño de la memoria intercalada a disco. F
- d) Es proporcional a la memoria RAM.

(Febrero 2008, Septiembre 2007, Septiembre 2006, Septiembre 2003)

25. El tamaño del TLB (table-lookaside buffer) en un sistema:

- a) Es proporcional al tamaño de la tabla de página de los procesos.
- **b) Depende del procesador.** Fuente: A3-8, L-349-355
- c) Es proporcional al tamaño de páginas.
- d) Es proporcional a la cantidad de procesos activos.
- a) Es variable dependiendo de la cantidad de RAM

(Febrero 2008, Febrero 2006 M1 y M2, Febrero 2005 -2 veces-, Septiembre 2004, Enero 2004 M2 y M1, Enero 2003, Febrero 2002)

9. Si en un sistema de ficheros de disco se usan bloques de mayor tamaño:

- **a) Disminuye la fragmentación externa. ¿?**
- b) Disminuye la fragmentación interna. F
- **c) Aumenta la fragmentación interna. ¿?**
- d) Aumenta la fragmentación externa. F
- a) Disminuye la fragmentación. F
- **b) El tamaño máximo de los ficheros aumenta exponencialmente. ¿?**
- **c) Aumenta la fragmentación interna. ¿?**
- d) Aumenta la fragmentación externa. F

- a) Disminuye la fragmentación. F
 - **b) Aumenta la velocidad de transferencia.**
 - c) Disminuye el rendimiento. F
 - d) Aumenta la fragmentación externa. F
-
- a) Disminuye la fragmentación. F
 - **b) Aumenta el rendimiento en cuanto a transferencia máxima de datos.**
 - c) Disminuye el rendimiento. F
 - d) Aumenta la fragmentación externa. F
-
- a) Disminuye la fragmentación. F
 - **b) Aumenta el tamaño máximo de los ficheros.**
 - c) Disminuye el rendimiento. F
 - d) Aumenta la fragmentación externa. F
-
- **a) El tamaño máximo de los ficheros aumenta linealmente** (si no utiliza i-nodos: FAT)
 - **b) El tamaño máximo de los ficheros aumenta exponencialmente** (si utiliza i-nodos)
 - c) Aumenta la fragmentación externa. F
 - d) Aumenta el espacio usado para mantener la meta información del sistema. F (depende del sistema de ficheros utilizado)

(Febrero 2007 M1 y M2)

19. Si se duplica el tamaño de los bloques en un sistema de ficheros basado en inodos:

- **a) Disminuye la fragmentación externa ¿?**
- b) Aumenta la fragmentación externa
- **c) El tamaño máximo de los ficheros aumenta exponencialmente**
- d) El tamaño máximo de los ficheros se duplica

(Septiembre 2005 M1 y M2)

19. Si en un sistema de ficheros de disco se usan bloques de menor tamaño:

- a) Disminuye la fragmentación externa.
- **b) Disminuye la fragmentación interna.**
- c) Aumenta la fragmentación interna.
- **d) Aumenta la fragmentación externa. ¿?**

(Febrero 2008, Septiembre 2007, Febrero 2006 M1 y M2, Septiembre 2005 M1 y M2)

10. Si se elimina el TLB (table-lookaside buffer) en un procesador:

- **a) El tiempo efectivo de acceso a memoria aumentará.** Fuente: A1-21, L-349
- b) El tiempo efectivo de acceso a memoria disminuirá.
- c) Será imposible implementar sistemas de páginas.

- d) Será imposible implementar sistemas de memoria virtual.
- **x) El tiempo efectivo de acceso será casi el doble**

(Febrero 2006 M1 y M2, Febrero 2002)

13. ¿Cuál de los siguientes privilegios de acceso UNIX (en octal) permite que un fichero pueda ser leído y ejecutado por cualquier usuario y modificado sólo por propietario y el grupo?

- a) 311
- b) 755
- c) 722
- **d) 775** read-write-execute y user-group-other: 111 111 101
- x) 711

(Septiembre 2004)

7. ¿Cuál de los siguientes privilegios de acceso UNIX (en octal) permite que un fichero pueda ser leído por cualquier usuario y modificado y ejecutado sólo por el propietario y el grupo?

- a) 772
- b) 755
- c) 722
- **d) 774** read-write-execute y user-group-other: 111 111 100

(Septiembre 2008, Febrero 2007 M1 y M3)

18) La señal SIGCHLD: (Pista: Si miras la función que usa el SIGCHLD del modelo básico de la práctica de SO (y seguramente habrás usado lo mismo para ASO) te darás cuenta.)

- a) Puede generar una llamada después que hayan acabado varios hijos.
- b) Genera una llamada por cada hijo que acaba.
- c) Puede generar una llamada aunque no haya acabado ningún hijo.
- d) Genera llamadas a la función indicada cada vez que acaba un proceso del mismo grupo.

(Septiembre 2002)

13. Un proceso en Unix crea N hijos enlazándolos con pipes, ¿cuántos wait deberán hacer los procesos en la tubería para sincronización y para asegurarse que todos los procesos del pipe terminan?

- a) Ninguno, la sincronización la hace el SO. Fuente: A3-8
- b) N.
- c) Igual al número de pipes (N-1).
- d) Depende del orden de creación.

(Septiembre 2007)

20. Un proceso Unix crea N hijos enlazándolos con tuberías, ¿cuántas señales SIGCHLD habrá recibido el padre cuando todos los hijos hayan terminado?

- a) Una sola, la que corresponda al último proceso de las tuberías
- b) N
- c) Igual al número de pipes (N-1)
- d) De 1 a N

(Febrero 2006 M1 y M2)

15. Se han creado varios procesos enlazados por tuberías. Si uno de ellos (cualquiera) todavía no terminó:

- a) Ninguno de los procesos puede haber terminado.
- b) Pueden haber terminado todos a excepción del último. Fuente: A3-3
- c) Pueden haber terminado todos a excepción del primero.
- d) Pueden haber terminado todos los posteriores.

(Septiembre 2007, Febrero 2006 M1 y M2)

16. ¿Qué método de lectura usa el SO cuando se hace un read sobre STDIN?

- **a) Por caracteres, canónico.**
 - Unidad de lectura o escritura es un carácter
 - No genera espera activa, leemos una línea entera
- b) Por caracteres, no canónico (raw).
- c) Por bloques, canónico (cooked).
- d) Por bloques, no canónico.

(Septiembre 2006, Febrero 2005, Septiembre 2004, Septiembre 2003, Septiembre 2002, Febrero 2002)

18. ¿Qué método de lectura usa el SO cuando se hace un fgets con stdin?:

- a) Por bloques, canónico (cooked).
- **b) Por caracteres, canónico.** Fuente: A3-6, Argumentación del anterior
- c) Por caracteres, no canónico (raw).
- d) Por bloques, no canónico.

(Febrero 2007 M1 y M2, Febrero 2006 M1 y M2, Septiembre 2004, Septiembre 2002, Enero 2002)

17. ¿Cuál de los siguientes campos NO se almacena en el i-nodo de un sistema de ficheros UNIX?

- a) Tamaño del fichero en bytes.
- **b) Nombre de fichero.** Fuente: A1-8, L-576, A3-2
- c) Permisos de acceso.
- d) Número de referencias.
- x) Fechas de acceso y modificación
- x) Hora de la última modificación

(Septiembre 2006, Febrero 2006 M1 y M2, Septiembre 2004, Septiembre 2003, Septiembre 2002, Febrero 2002)

18. ¿Cuál es la ventaja de un sistema de ficheros con journaling con respecto a uno que no lo tenga?

- a) Asegura que los datos en el disco son consistentes en todo momento. F
- **b) Tiempos de recuperación lineales en caso de caída.** Fuente: A0-2
- c) Tiempos de recuperación exponenciales en caso de caída. F
- d) Minimiza la dispersión de bloques de un fichero. F
- a) Es más rápido en las escrituras.
- **b) Asegura la consistencia del sistema de ficheros en caso de corte imprevisto en el suministro eléctrico.**
- c) Más rápido en las lecturas.
- d) Mejor gestión de sistemas de buffers y cache.
- **b) Posibilidad de controlar las “transacciones”.**
- **b) Recuperación de datos de disco más rápida si el sistema "cae" imprevistamente**

(Septiembre 2007)

7. ¿Qué podría anular las ventajas de un sistema de ficheros con journaling?

- a) Caídas no programadas del sistema F
- b) Muchos datos de usuarios pendientes de grabar
- c) Caché interna de discos duros
- d) Sistemas de buffers y cache de disco

(Septiembre 2006, Febrero 2006 M1 y M2, Septiembre 2003, Septiembre 2002)

19. ¿Cuál de los siguientes NO es un sistema de ficheros con journaling?

- a) ReiserFS
- b) NTFS
- **c) Ext2**
- d) XFS
- **b) FAT32**
- c) Ext3
- **b) UFS (Unix File System)**

(Enero 2009, Febrero 2007 M1)

2. ¿Dónde puede escribirse el “diario” en un sistema de ficheros con journaling?

- a) En la memoria RAM
- b) En los bloques del mismo dispositivo donde está el sistema de ficheros
- **c) En cualquier memoria no volátil del ordenador**
- d) En un dispositivo independiente y diferente al que mantiene el sistema de ficheros

(Enero 2009)

11. ¿Cuál de las siguientes razones ayuda a que el vaciado (flush) del TLB es tan “costoso”?

- a) Por que hay que leer entradas nuevas de las tablas de páginas

- b) Porque hay que intercalar las páginas al dispositivo de almacenamiento secundarios
- c) Por el algoritmo de selección de las víctimas
- **d) Porque hay que copiar los datos modificados a las tablas de página**

(Febrero 2007 M1 y M2)

6. ¿Qué componente hardware de los procesadores es el que más afecta a la tasa de aciertos de la tabla de páginas?

- a) Caché de memoria
- b) Tamaño de las tablas de página
- c) TLB
- d) Tamaño de los marcos

(Febrero 2007 M1 y M2)

11. ¿Qué haría para aumentar la tasa de aciertos del TLB en un procesador?

- a) Disminuir el tamaño de las páginas
- b) Aumentar el tamaño de las páginas
- c) Aumentar la memoria cache
- d) Usar un algoritmo de reemplazo que no sufra la anomalía de Belady

(Febrero 2006 M1 y M2)

20. La tasa de aciertos o hit ratio de un TLB

- a) Es proporcional a la velocidad del procesador.
- b) Es proporcional a la velocidad de acceso a memoria RAM.
- **c) Es inversamente proporcional al tamaño del conjunto residente del proceso.** Fuente: L-354
- d) Es proporcional a la cantidad total de memoria RAM.

(Septiembre 2007, Septiembre 2006, Febrero 2005, Septiembre 2004, Enero 2004 M2 y M1, Septiembre 2003, Enero 2003)

2. En un sistema de memoria virtual con paginación se observa que la tasa de aciertos del TLB es muy baja. ¿Que debería hacerse?

- a) Si es posible, disminuir el tamaño de las páginas.
- b) Aumentar el tamaño del conjunto de trabajo de los procesos.
- c) Disminuir el tamaño del conjunto de trabajo de los procesos.
- **d) Si es posible, aumentar el tamaño de las páginas.** Fuente: L-353, A3-6

(Enero 2009, Septiembre 2007, Septiembre 2006, Febrero 2006 M1 y M2, Febrero 2005, Septiembre 2004, Enero 2004 M2 y M1, Septiembre 2003, Enero 2003, Septiembre 2002, Enero 2002)

22. El algoritmo del reloj (CLOCK) para reemplazo de páginas:

- a) Es la implementación del LRU.
- **b) Es una aproximación del LRU.** Fuente: A3-4, L-364, L-385, A0-30

- c) Ordena la lista de páginas dependiendo de la frecuencia de uso.
- d) Sufre de la anomalía de Belady.
- a) Es similar al FIFO.
- b) Es similar al LIFO.
- c) Ordena a las páginas por su el tiempo transcurrido desde el último acceso.
- **d) Es el algoritmo de PILA de segunda oportunidad.** Fuente: A1-25, A0-26
- **a) Beneficia a las páginas recientemente utilizadas.** L-364, A3-4
- b) Ordena la lista de páginas dependiendo de la frecuencia de uso. F
- c) Asigna un período de tiempo predefinido adicional a la página para ver si es utilizada en ese período. F
- d) Beneficia a la página menos recientemente utilizada.
- **a) Es similar al FIFO de segunda oportunidad.** Fuente: A1-25
- b) Ordena a las páginas por su frecuencia de uso.
- c) Asigna un período de tiempo (predefinido) adicional a la página para ver si es utilizada en ese período.
- **a) Es una buena aproximación al LRU.** Fuente: A3-4, L-364, L-385, Examen
- b) Ordena la lista de páginas dependiendo de la frecuencia de uso.
- c) Asigna un período de tiempo (predefinido) adicional a la página para ver si es utilizada en ese período.
- a) Sólo puede usarse para el reemplazo de páginas en gestión de memoria virtual
- b) Podría usarse también para selección de víctimas en sistemas de buffer-cache
- c) Ordena la lista de páginas dependiendo de la frecuencia de acceso. F
- d) Asigna un periodo de tiempo predefinido adicional a la página para ver si es utilizada en ese periodo, F

(Febrero 2007 M1 y M2)

7. ¿Qué técnica que mejor aproxime al LRU usaría para el reemplazo de páginas o bloques en un sistema de buffer-caché de dispositivos de bloques?

- a) FIFO
- **b) CLOCK ¿?**
- c) LRU
- d) SCAN

(Febrero 2008, Septiembre 2005 M1 y M2)

1. Si se implementa una sola sección crítica (mutex o lectores-escriptores) en la práctica. ¿Dónde debería accederse a dicha sección crítica en la función `mi_write()` de la forma más óptima?

- a) Al principio de la función en `directorio.c`
 - b) Al principio de la función en `ficheros.c`
 - c) En el agente, apenas se recibe el mensaje.
 - **d) Sólo si hay que reservar nuevos bloques.**
-
- a) Al principio de la función en `directorio.c`
 - b) Al principio de la función en `ficheros.c`
 - c) No hace falta exclusión mutua en el `mi_write()`
 - **d) Sólo en el caso que deban asignarse nuevos bloques al fichero**

(Enero 2004 M2 y M1)

11. Si se implementa una sola sección crítica (mutex o lectores-escriptores) en la práctica. ¿Dónde debería accederse a dicha sección crítica en la función `mi_write()`?

- a) El principio de la función en `directorio.c`
- b) Al principio de la función en `ficheros.c`
- c) En el agente, apenas se recibe el mensaje.
- **d) Sólo en la parte del código que modifica la meta-información.**

Fuente: A3-10, A0-9

(Febrero 2008)

16. ¿Cuántos semáforos distintos se necesitan para la sección crítica anterior?

- **a) 1 ¿?**
- b) 2
- c) Uno por cada función que se llama desde `directorio.c`
- d) Se crean dinámicamente al llamarse las funciones de `directorio.c`

(Febrero 2008)

17. Para el mismo caso anterior, la función `mi_read()` no es bloqueante, si un programa está creando un fichero, otro está a punto de borrar un fichero de otro directorio y hay varios lectores

- a) Todos se bloquearán hasta que terminen los que crean o borran ficheros
- b) Todos se ejecutarán de forma concurrente
- c) Se bloquean los lectores y el que borra se ejecutará concurrentemente
- d) Se bloqueará el que borra, los lectores continuarán “asincrónamente”

(Febrero 2005)

3. Con una sola sección crítica (lectores-escriptores) en la práctica. ¿Dónde debería accederse a dicha sección crítica en la función `mi_write()`?

- a) El principio de la función en `directorio.c`

- b) Al principio de la función en ficheros.c
- c) En cuanto se recibe el mensaje.
- **d) Sólo en el caso que deba asignarse nuevos bloques.**

(Septiembre 2005 M1 y M2, Enero 2004 M2 y M1)

2. El tamaño del fichero que almacenamos en el inodo (y se devuelve con el `mi_stat()`):

- **a) Indica la posición del último byte escrito.**
- ~~b) Indica el número real de bytes escritos en el fichero.~~ Fuente: A3-9
- c) Indica el número bloques asignados por el tamaño en bytes de cada bloque.
- d) Indica el número bloques teóricos usados por el tamaño en bytes de cada bloque.

(Septiembre 2005 M1 y M2, Febrero 2005, Enero 2004 M2 y M1)

3. La función `mi_dir()` implementada en `directorio.c` (o equivalente):

- a) Recorre el inodo del directorio para leer los nombres de fichero.
 - b) Recorre sólo los bloques de datos del directorio.
 - c) Llama a la función de directorios del `ficheros.c` (o equivalente).
 - **d) Sólo hace un recorrido secuencial del fichero del directorio usando el `mi_read()` de `ficheros.c`.** Fuente: A3-9
-
- a) Sólo lee el i-nodo correspondiente al directorio para leer los nombres de fichero.
 - b) Además de leer el i-nodo, podría leer también los índices secundarios.
 - c) Llama a la función de directorios del `ficheros.c`.
 - **d) Hace un recorrido secuencial del fichero del directorio usando el `mi_read()` de `ficheros.c`.** Fuente: A3-9

(Septiembre 2005 M1 y M2)

5. Si sólo se implementa una sección crítica (mutex o lectores-escritores) en los servidores de la práctica. ¿Cuántos clientes podrían estar creando ficheros “simultáneamente”?

- **a) 1 ¿?**
- b) 2
- c) Uno por cada proceso servidor.
- d) Sin límites.

(Enero 2004 M2 y M1)

14. Con un sólo semáforo mutex en los servidores de la práctica. ¿Cuántos clientes podrían estar creando ficheros “simultáneamente”?

- **a) 1** Fuente: A3-9
- b) 2
- c) Uno por cada proceso servidor.
- d) Sin límites.

(Febrero 2005)

5. ¿Cuántos clientes podrían estar creando ficheros “simultáneamente” en la práctica? (con control vía lectores-escritores).

- a) 1 ¿?
- b) 2
- c) Uno por cada proceso servidor.
- d) Sin límites.

(Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Septiembre 2003, Enero 2003)

6. Las operaciones de apertura y cierre del “sistema de ficheros en disco” (open y close) en la implementación de la práctica:

- a) Se llaman cada vez que se hace una operación de lectura escritura de uno o varios bloques.
- b) Se llaman cada vez que se intenta leer o escribir un bloque.
- **c) Se ejecutan una sola vez, al inicio del servidor o cuando se “monta” el sistema de ficheros.** Fuente: A3-8
- d) Es indistinto.

(Febrero 2008)

20. En un programa que use las “librerías” de la práctica, éste monta el sistema de ficheros y hace cientos de operaciones antes de salir. ¿Cuántas veces se habrá llamado al open() desde bloques.c?

- a) Una ¿?
- b) Una por cada llamada a una función de directorios
- c) Puede llamarse más de una vez por cada llamada a una función de directorio.c
- d) Una cada vez que se lee o escribe un bloque

(Septiembre 2005 M1 y M2)

7. ¿Para qué se usa memoria compartida en la práctica?:

- a) Sincronización entre servidores.
- b) Para mantener el super bloque y el mapa de bits.
- c) Para control los accesos y concurrencia.
- d) Sólo hace falta si se implementan técnicas de buffer-cache. Fuente:

A3-9

(Enero 2004 M2 y M1, Septiembre 2003)

17. En la práctica, el uso de memoria compartida:

- a) Es necesaria para la sincronización entre servidores.
- b) Es imprescindible para mantener el super bloque y el mapa de bits.

- c) No es necesaria si se hacen los accesos y control de concurrencia correctos. Fuente: A3-9
- d) Se necesita siempre porque se implementan técnicas de buffer-cache obligatoriamente.
- a) Siempre es necesaria para la sincronización entre servidores.
- b) Sólo si se usan técnicas de lectores-escritores para el control de concurrencia.
- c) Si se usa mmap en vez de read y write.
- d) Sólo es necesaria si se implementan técnicas de buffer-cache. Fuente: A3-9

(Enero 2003)

1. En la práctica es imprescindible el uso de memoria compartida:

- a) Siempre es necesaria para la sincronización entre servidores.
- b) Sólo si se usan técnicas de lectores-escritores para el control de concurrencia.
- c) Si se usa mmap en vez de read y write.
- d) Sólo es necesaria si se implementan técnicas de buffer-cache. Fuente: A3-9

(Septiembre 2005 M1 y M2, Enero 2004 M2 y M1)

8. ¿Que método de control de concurrencia en los servidores funcionaría mejor a efectos de maximizar el “paralelismo” de operaciones?

- a) Una sección crítica con un mutex.
- b) Una sección crítica con lectores-escritores.
- c) Varias secciones críticas con un mutex cada una.
- d) Varias secciones críticas con lectores-escritores. Fuente: A3-9

(Enero 2003)

3. Si cada cliente de la práctica hiciese, además de sólo escrituras, lecturas y escrituras sobre los ficheros. ¿Que método de control de concurrencia en los servidores funcionaría mejor?

- a) Una sección crítica con un mutex.
- b) Una sección crítica con lectores-escritores. Fuente: A3-9
- c) Varias secciones críticas con un mutex cada una.
- d) Varias secciones críticas con lectores-escritores.

(Septiembre 2005 M1 y M2, Enero 2004 M2 y M1)

9. Una sola operación de escritura con el `mi_write()` (sin buffer-cache):

- a) Podría generar la escritura de hasta dos bloques completos. Fuente: A3-9
- **b) Podría generar la lectura de hasta dos bloques.**
- c) Podría generar la lectura de más de dos bloques.

- d) Sólo se escribe.

(Enero 2009, Febrero 2008, Febrero 2005)

7. Una sola operación de escritura con el `mi_write()`: (Se refiere a bloques de datos, no `metainfo`.)

- a) Podría generar la escritura de hasta dos bloques completos.
- **b) Podría generar la lectura de hasta dos bloques.**
- c) Podría generar la lectura de más de dos bloques.
- d) Sólo se escribe.

(Septiembre 2003, Enero 2003)

19. Si un cliente de la práctica hace una operación de escritura de sólo 2 bytes (sin `buffer-cache`):

- a) Podría generar la escritura de hasta dos bloques completos.
- **b) Podría generar la lectura y escritura de dos bloques completos.**
- c) Se lee y escribe sólo un bloque. F
- d) Sólo se escribe. F

(Enero 2009)

18. En la práctica, ¿en qué operaciones se puede producir una asignación de bloques de datos a un fichero?

- a) En la creación del fichero al momento de asignar un inodo
- **b) Sólo en la escritura de datos**
- c) En la escritura y el `truncate`
- d) En la creación y/o la escritura

(Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Enero 2003)

10. La arquitectura general de los procesos de la práctica es:

- a) Lectores/escritores.
- b) Cliente-servidor, servidores con estado.
- c) Cliente-servidor, servidores sin estado. Fuente: A3-9
- d) Ninguna de las demás opciones.

(Septiembre 2005 M1 y M2)

11. En la práctica, si se implementa el control de concurrencia de escrituras a un fichero, con una semántica de consistencia similar al Unix/Linux, ¿dónde?:

- a) En los clientes.
- b) En los servidores.
- c) En ninguno.
- d) Hay que hacerlo tanto en los clientes como en los servidores.

(Enero 2004 M2 y M1)

22. Si hubiese que implementar control de concurrencia a los datos de un fichero con una semántica de consistencia similar al Unix/Linux:

- a) Hay que hacerlo en los clientes.
- b) Es más sencillo hacerlo en los servidores.
- c) Es indistinto.
- d) Hay que hacerlo tanto en los clientes como en los servidores.

(Septiembre 2004, Enero 2003)

18. Si hubiese que implementar control de concurrencia a los datos de un fichero:

- a) Es más sencillo hacerlo en los clientes. Fuente: A3-10
- b) Es más sencillo hacerlo en los servidores.
- c) Es indistinto.
- d) Hay que hacerlo tanto en los clientes como en los servidores.

(Enero 2009, Febrero 2008)

5. En la práctica, si los programas necesitan controlar control de concurrencia a los datos de un fichero:

- a) No hace falta, ya lo proveen las librerías del sistema de ficheros
- **b) Se deben implementar sus propias secciones críticas y/o semáforos a “nivel de aplicación”**
- c) Habría que implementar controles de concurrencia adicionales en los programas y también en las librerías
- d) Sólo se puede hacer a través de mecanismos provistos por el sistema de ficheros

(Septiembre 2005 M1 y M2, Enero 2004 M2 y M1, Enero 2003)

12. ¿Dónde se almacena la información de privilegios de acceso a los ficheros para el propietario, el grupo y “otros” (similar al UNIX) en el sistema de ficheros de la práctica?

- a) En la entrada del fichero en el directorio.
- b) Se necesita un área especial en el sistema de ficheros.
- **c) En el inodo del fichero.** Fuente: A3-9
- d) Se debe almacenar en la información de usuario, no en el sistema de ficheros.

(Febrero 2005, Septiembre 2004, Enero 2004 M2 y M1, Septiembre 2003, Enero 2003)

6. Se desea que la librería de clientes de la práctica permitiese leer y escribir buffers de tamaño “indefinido”, es decir, sin límite predeterminado:

- a) No se puede implementar de la forma en que está definida la arquitectura de la práctica.
- b) Se puede, pero estará limitado por la cantidad de memoria disponible y se podrían generar varios mensajes por cada operación. Fuente: A3-9
- c) Es posible, y se puede enviar todo el buffer en un sólo mensaje.

- d) Desde el punto de vista de las librerías para los clientes, el tamaño del buffer está predefinido.

(Febrero 2007 M1 y M2, Febrero 2005)

8. ¿Cuál de las siguientes diferencias entre i-nodos y FAT es verdadera?

- a) FAT favorece a los accesos aleatorios.
- b) Los i-nodos son más adecuados para accesos aleatorios. F
- c) Los i-nodos penalizan a los ficheros pequeños. F
- d) Las tablas FAT favorecen el acceso a ficheros más grandes. Fuente: A1-5, A3-5

(Enero 2004 M2 y M1, Enero 2003)

24. ¿Cuál es la ventaja de los sistemas de ficheros de inodos con respecto al FAT?

- a) El FAT requiere menos memoria para minimizar los accesos. F? Fuente: L-578
- b) Los inodos se adecuan más a ficheros y sistemas de ficheros de mayor tamaño. Fuente: A1-5
- c) Los inodos penalizan a los ficheros pequeños. F
- d) Las tablas FAT favorecen el acceso a ficheros más grandes.

(Febrero 2005, Enero 2004 M2 y M1)

12. Si se aumenta el tamaño del TLB:

- **a) El tiempo efectivo de acceso a memoria debería aumentar.** Fuente: L-354, A3-8
- b) El tiempo efectivo de acceso a memoria debería disminuir.
- c) Deberían disminuir los fallos de páginas.
- d) Deberían aumentar los fallos de páginas.

(Febrero 2005)

13. ¿Cuál es el método mas sencillo desde el punto de vista del programador para la lectura y escritura a una terminal de usuario en modo texto?

- **a) Modo canónico.** Fuente: A3-10, A0-12
- b) Modo no canónico.
- c) Curses.
- d) Acceso directo al dispositivo.

(Septiembre 2007, Septiembre 2006, Septiembre 2003, Enero 2003)

24. ¿Cuál es el método más flexible implementado en un sistema UNIX y que permite controlar, en modo texto, con mayor precisión la actividad y estado del teclado?

- a) Modo canónico (cooked).
- **b) Modo no canónico (raw).** Fuente: A3-6, A0-12
- c) Curses.

- d) Acceso directo al dispositivo.

(Febrero 2005, Septiembre 2003, Septiembre 2002)

14. En el sistema POSIX de mensajes:

- a) Los mensajes deben ser de un tamaño fijo.
- b) Se deben definir para cada cola que se crea.
- c) Se deben definir al usar por primera vez una cola.
- d) Pueden variar de tamaño. Fuente: A3-10

(Febrero 2006 M1 y M2)

21. El sistema de ficheros que utiliza el sistema de i-nodos estándar, con un tamaño de bloque de 4096 bytes, ¿cuántos bloques de disco se deberán leer en total para obtener el byte que está en la posición 1024^3+1 del fichero? (*el i-nodo está leído y en memoria*).

- a) 2.
- b) 3.
- **c) 4.**
 - $(1024^3+1)/4096 = 262.114 > 10+1024^2 \mid (1024^3+1)/4096/4096 = 64$
 - $4096/4 = 1024 \mid 10+1024+1024^2+1024^3 > 262.114 >$
- $10+1024+1024^2 = \text{Nivel 3}$
- d) Más de 4.

(Febrero 2005)

16. El sistema de ficheros que utiliza el sistema de i-nodos de Unix estándar (con 13 índices en total, los últimos tres son indirectos), con un tamaño de bloque de 4096 bytes, ¿cuántos bloques deberán leerse en total (*incluido la lectura del primer i-nodo*) para obtener el byte que está en la posición 150.000 del fichero?

- a) 1.
- b) 2.
- **c) 3.**
 - $150000/4096 = 36, \mid 1 \text{ bloque primer inodo} + 1 \text{ bloque indirecto} + 1$
- $\text{bloque directo (36)}$
- $4096/4 = 1024 \mid 10+1024 > 36 > 10 = \text{Nivel 1}$
- d) Más de 3.

(Septiembre 2004)

8. El sistema de ficheros que utiliza el sistema de i-nodos (con 13 índices en total, los últimos tres son indirectos), con un tamaño de bloque de 4096 bytes, ¿cuántos i-nodos deberán leerse en total (*incluido la lectura del primer i-nodo*) para obtener el byte que está en la posición 25.000 del fichero?.

- a) 1.
- **b) 2.**
 - $25000/4096 = 6, \mid 1 \text{ bloque directo (el 6)} + 1 \text{ del bloque del inodo inicial}$
- c) 3.
- d) Más de 3.

(Febrero 2002)

9. El sistema de ficheros que utiliza el sistema de i-nodos (con 15 índices en total, los últimos tres son indirectos), con un tamaño de bloque de 4096 bytes, ¿cuántos i-nodos deberán leerse en total (*incluido la lectura del primer i-nodo*) para obtener el byte que está en la posición 50.000 del fichero?.

- a) 1.
- **b) 2.**
- $50000/4096 = 12, |$ 1 bloque directo (el 12) +1 del bloque del inodo inicial
- c) 3.
- d) Más de 3.

(Febrero 2005)

17. Los fabricantes de un microprocesador aseguran que con el TLB obtienen un hit-ratio del 98%. Si el tiempo de respuesta del TLB es despreciable y el tiempo de acceso a memoria es de 10 ns. ¿Cuál es el tiempo efectivo de acceso?

- **a) 10.2** $= \alpha \cdot t_a + 2 \cdot (1-\alpha) \cdot t_a + t_r = 0,98 \cdot 10\text{ns} + 2 \cdot (1-0,98) \cdot 10\text{ns} +$
despreciable
- b) 10.5
- c) 11.2
- d) 10.8

(Septiembre 2004, Febrero 2002)

11. Los fabricantes de un microprocesador aseguran que se obtiene un hit-ratio del 95% utilizando la memoria caché asociativa (TLB). Si el tiempo de respuesta del TLB es despreciable y el tiempo de acceso a memoria es de 10 ns. ¿Cuál es el tiempo efectivo de acceso?

- a) 10.2
- **b) 10.5** $= 0,95 \cdot 10\text{ns} + 2 \cdot (1-0,95) \cdot 10\text{ns} +$ despreciable
- c) 11.2
- d) 10.7
- b) 20
- c) 11.5
- d) Ninguno de los anteriores.

(Septiembre 2007)

18. Los fabricantes de un microprocesador aseguran que se obtiene un hit-ratio del 95% utilizando la memoria caché asociativa (TLB). Si el tiempo de respuesta del TLB es 2ns y el tiempo de acceso a memoria es de 10 ns. ¿Cuál es el tiempo efectivo de acceso? (tomar en cuenta el tiempo de respuesta del TLB)

- a) 11
- b) 11,5

- c) 10,5
- **d) $12,5 = 0.95*10 + 2*(1-0.95)*10 + 2$**

(Septiembre 2004, Febrero 2002)

1. En un sistema multiusuario y multiprogramación con conjuntos de trabajo, uno de los procesos (con un conjunto residente relativamente pequeño) genera muchas fallas de página. ¿Qué debería hacerse?

- a) Estudiar y quizás reorganizar la estructura del programa.
- b) Habría que disminuir el conjunto residente del proceso.
- c) Habría que aumentar el conjunto residente del proceso. Fuente: A3-1
- d) Disminuir la cantidad de procesos activos.

(Septiembre 2003, Septiembre 2002)

9. En un sistema multiusuario y multiprogramación con conjuntos de trabajo, uno de los procesos (que no usa demasiada memoria) genera muchas fallas de página. ¿Qué debería hacerse?

- a) Estudiar y quizás reorganizar la estructura del programa.
- b) Habría que disminuir el área activa del proceso.
- c) Habría que aumentar el área activa del proceso.
- d) Disminuir la cantidad de procesos activos.

(Septiembre 2006)

7. En un sistema multiusuario y multiprogramación con conjuntos de trabajo, uno de los procesos que ya tiene un conjunto de trabajo relativamente grande genera muchas fallas de página. ¿Qué debería hacerse?

- a) Estudiar y quizás reorganizar la estructura del programa.
- b) Habría que disminuir el área activa del proceso.
- c) Habría que aumentar el área activa del proceso.
- d) Disminuir la cantidad de procesos activos.

(Enero 2009, Febrero 2008, Septiembre 2007)

17. En un sistema multiusuario y multiprogramación con conjuntos de trabajo, ¿qué debería hacer el gestor de memoria virtual si uno de los procesos entra en la fase de migración de conjunto de trabajo?

- a) Reducir el área activa
- b) Reducir el número de procesos activos
- **c) Aumentar el área activa del proceso**
- d) Aumentar el número de procesos activos

(Enero 2009, Febrero 2007 M1 y M2)

19. La técnica de “conjuntos de trabajo” usa técnicas fundamentalmente basadas en:

- a) LRU

- b) CLOCK
- c) Fallos del TLB
- **d) Frecuencias de fallos de página**
- **d) Análisis de frecuencias de fallos de página**

(Septiembre 2004, Enero 2002)

2.El uso de esquemas cliente/servidor para comunicaciones.

- a) Uno de los procesos puede leer y el otro escribir.
- b) El servidor no almacena información relativa al cliente.
- c) El servidor es siempre el que envía datos.
- d) Uno de ellos inicia la comunicación. Fuente: A3-10

(Enero 2009, Septiembre 2004, Septiembre 2002, Enero 2002)

5.Los sistemas operativos con memoria virtual se diseñaron básicamente para:

- a) Permitir que el espacio físico de direccionamiento sea mayor que el espacio lógico. F
- b) Mejorar los tiempos de respuesta en las operaciones de E/S. F
- c) Independizar el espacio lógico de direccionamiento del espacio físico.

Fuente: [Internet](#)

- **d) Permitir que los procesos inactivos no ocupen memoria RAM.**
- x) Aumentar el uso de la CPU aumentando la cantidad de procesos activos Fuente: [Internet](#)
- x) Balancear mejor el uso de CPU de cada proceso F
- **x) Permitir que el espacio lógico de direccionamiento sea mayor que el espacio físico**

(Septiembre 2004, Septiembre 2003, Febrero 2002)

10.Los descriptores de ficheros que retorna la función open:

- a) No son heredadas por los hijos del proceso.
- b) Son heredadas por los hijos, pero sólo puede haber un escritor.
- c) Son heredados por todos y pueden hacer sólo las mismas operaciones que podía hacer el padre. Fuente: A3-6
- d) Son heredadas por todos y pueden ser usados para cualquier operación.

- a) No son heredadas por los hijos del proceso.
- b) Son heredadas por los hijos, pero sólo uno puede acceder.
- c) Son heredadas por los hijos, cualquiera de ellos puede leer o escribir.

Fuente: A3-6

- d) Sólo es heredada por el primer hijo, que puede ser el lector o escritor.

(Septiembre 2006, Enero 2003)

21. Los descriptores de ficheros que retorna la función open:

- a) Sólo pueden ser usadas para E/S a dispositivos de bloques.
- b) Pueden ser usadas con las funciones fopen y fwrite.
- c) Mantienen buffers en memoria del proceso para minimizar la E/S.

Fuente: A3-6

- d) Puede representar a un dispositivo de bloque, carácter o de red.
- a) No sirven para la consola, sólo pueden ser usadas para E/S a dispositivos de bloque
- b) Pueden ser usadas con las funciones fopen y fwrite
- c) Mantienen buffers en memoria del proceso para minimizar la E/S
- d) Son números enteros

(Septiembre 2007, Septiembre 2003)

23. Los descriptores de ficheros que retorna la función fopen:

- a) No sirven para la consola, sólo pueden ser usadas para E/S a dispositivos de bloques.
- b) Pueden ser usadas con las funciones open y write.
- c) Mantienen buffers en memoria del proceso para minimizar la E/S.

Fuente: A3-6 ¿?

- d) Puede representar a un dispositivo de bloque, carácter o de red.
- x) Son del tipo entero (int)
- a) No son heredadas por los hijos del proceso
- b) Son heredadas por los hijos, pero sólo uno puede acceder
- c) Son heredadas por los hijos, cualquiera de ellos puede leer o escribir
- d) Sólo es heredada por el primer hijo, que puede ser el lector o escritor

(Septiembre 2002)

16. Los descriptores de ficheros que retorna la función pipe:

- a) No son heredadas por los hijos del proceso.
- b) Son heredadas por los hijos, pero sólo puede haber un escritor y un lector. Fuente: A3-3
- c) Son heredadas por los hijos, puede haber cualquier número de lectores y escritores.
- d) Sólo es heredada por el primer hijo, que puede ser el lector o escritor.

(Septiembre 2004, Enero 2003)

20. En la práctica un cliente hace una operación de escritura que genera varios mensajes hacia los servidores:

- a) Es imposible, no puede funcionar.
- b) Deben ser atendidos por el mismo servidor para asegurar la consistencia del sistema de ficheros. Fuente: A3-10
- c) Pueden ser atendidos por cualquier servidor.
- d) Los servidores deben mantener información de estado de los ficheros.

(Septiembre 2007, Septiembre 2006, Septiembre 2003, Septiembre 2002)

1. En un sistema con paginación por demanda:

- **a) Se producen muchas fallas de páginas durante la carga del programa.** Fuente: L-361, A3-3
- b) El objetivo es aumentar la velocidad de ejecución de los programas a costa de disminuir el nivel de multiprogramación.
- c) La cantidad de procesos activos depende exclusivamente de la memoria RAM disponible.
- d) Las fallas de páginas son constantes durante toda la ejecución de los programas.
- x) Se producen pocos fallos de páginas durante la carga del programa

(Febrero 2007 M1 y M2)

13. Al final de la ejecución de un proceso en un sistema con paginación por demanda, su conjunto residente será:

- a) Siempre igual al total de páginas del proceso
- b) Siempre menor
- c) Igual o menor al total de páginas del procesos
- d) Igual al tamaño máximo de las tablas de páginas

(Febrero 2008, Septiembre 2007, Septiembre 2006, Septiembre 2003, Septiembre 2002)

2. Para evitar tener que copiar todo el espacio de direcciones del padre cuando se ejecuta un fork, la mayoría de los sistemas UNIX implementan la técnica copy-on-write. ¿Cuál de los siguientes casos es el más favorecido?

- a) Es indistinto.
 - **b) Cuando el hijo hace pocas modificaciones a los datos en memoria.**
- Fuente: A3-5, A0-29
- c) Cuando el hijo hace más modificaciones a los datos en memoria.
 - d) Cuando la memoria residente del padre es relativamente pequeña.

- a) Es indistinto. F
- b) Cuando el hijo comparte memoria del padre.
- c) Cuando la memoria residente del padre es relativamente grande.
- d) Cuando la memoria residente del padre es relativamente pequeña. F

- a) Es indistinto F
- **b) Cuando el hijo ejecuta un exec...() inmediatamente.** Eliminación
- c) Cuando el hijo hace más modificaciones a los datos en memoria F
- d) Cuando la memoria residente del padre es relativamente pequeña F

(Septiembre 2006, Septiembre 2003, Septiembre 2002)

3. ¿Qué información se puede obtener de la llamada de sistema wait?

- `pid_t wait(int *status);`
- `pid_t waitpid(pid_t pid, int *status, int options);`

- a) Error del sistema si existe, valor de salida y PID del hijo. Fuente: A3-6
- b) Valor de salida y PID del hijo.
- c) Error del sistema si existe y valor de salida del hijo.
- d) Valor de salida del hijo.

(Febrero 2007 M1 y M2)

18. ¿Qué información se puede obtener de la llamada de sistema wait3/wait4?

- `pid_t wait3(int *status, int options, struct rusage *rusage);`
- `pid_t wait4(pid_t pid, int *status, int options, struct rusage *rusage);`
- a) Error del sistema si existe, valor de salida y PID del hijo
- b) Valor de salida y PID del hijo.
- c) Error del sistema si existe y valor de salida del hijo.
- d) PID del hijo, valor de salida y estadísticas de uso de recursos (rusage)

(Septiembre 2003, Septiembre 2002)

4. En el comando `mi_fsck`, de la práctica:

- a) Puede hacer a través del sistema de mensajes con los servidores.
- b) Sólo puede hacerlo accediendo directamente al sistema de ficheros. ¿?

Fuente: A3-10

- c) Es indistinto, puede hacerlo de las dos maneras.
- d) ¿Que es el `mi_fsck`?

(Septiembre 2003, Septiembre 2002)

5. ¿Cuál es la forma más flexible de implementar el sistema de directorios en la práctica?

- a) Mediante una tabla fija, con bloques reservados.
- b) Con tabla dinámica sobre bloques reservados en el sistema de ficheros.
- **c) Con los datos almacenados como si fuesen ficheros.** Fuente: A3-9, Prácticas.odt
- d) En la práctica no implementamos un sistema de directorios.

(Enero 2009)

3. De las siguientes, ¿cuál es el principal problema de rendimiento del sistema de directorios de la práctica?

- a) Ser tratado como un fichero
- b) No almacenar más propiedades de los ficheros en el propio directorios
- c) Hay que mantener en cache el directorio actual de trabajo
- **d) El tiempo de búsqueda**

(Septiembre 2006, Septiembre 2003, Septiembre 2002)

8. En UNIX, si un proceso acaba por error de ejecución y tiene varios hijos activos, ¿qué pasa con ellos?

- a) Finalizan la ejecución inmediatamente.

- b) El sistema los marca como hijos del init y continúan su ejecución normalmente. Fuente: A3-5
- c) El padre permanece en estado zombie hasta que todos los hijos terminen.
- d) Los hijos quedan en estado zombie o defunct.

(Septiembre 2006, Septiembre 2003, Septiembre 2002, Febrero 2002)

11. Una llamada de sistema en UNIX:

- a) Está implementada totalmente en el espacio de usuario.
- b) Produce una interrupción de hardware.
- **c) Genera una interrupción por software (trap).** Fuente: A3-5
- d) Es una llamada a una función del kernel mediante un cambio en el registro de contador de programa del procesador.
- **c) Genera una llamada a una función del kernel mediante una interrupción de software.** Fuente: A3-5

(Septiembre 2007, Septiembre 2006, Septiembre 2003, Septiembre 2002)

13. Indique cuál de las opciones es falsa para el sistema operativo Linux/para el núcleo Linux:

- a) Permite compartir páginas de memoria.
- **b) Tiene arquitectura de micro-kernel.** Fuente: A3-7, Eliminación, Wikipedia
- c) Intenta utilizar el máximo de memoria RAM disponible para buffer/cache de disco.
- d) Cualquier interacción entre procesos y el SO es a través de llamadas de sistema.
- **b) No permite la carga dinámica de módulos y librerías**
- a) Permite compartir páginas de memoria F
- b) Permite la carga dinámica de módulos y librerías de usuarios F
- c) Usa la paginación por demanda F, por ejemplo: mmap
- **d) Los módulos del núcleo deber cargarse al arranque del sistema ¿?**

(Enero 2003)

6. En la práctica se quiere maximizar el tamaño máximo de cada fichero en el sistema de ficheros:

- a) Usaría los mismos inodos ya reservados como índices indirectos.
- **b) Almacenaría los índices indirectos en los bloques de datos, o aumentaría el tamaño de los bloques.** Fuente: A3-9
- c) El tamaño máximo está sólo determinado por la cantidad de bits que se usan en los índices.
- d) No hay diferencia, a menos que se aumente la cantidad de punteros indirectos en los inodos.

(Febrero 2002)

7. La anomalía de Belady muestra:

- a) Que para algunos algoritmos de reemplazo de páginas (local), si se aumenta la cantidad de marcos asignados, disminuyen las fallas de páginas.
- b) Que existe un algoritmo de reemplazo de páginas que asegura la mínima cantidad de fallas de página.
- c) Que conociendo la información del pasado de un proceso es posible saber cuál será la página que no se usará durante el período más largo.
- **d) Que para algunos algoritmos de reemplazo de páginas, si se aumenta la cantidad de marcos asignados, no disminuyen los fallos de páginas.** Fuente: A3-10, A1-27, A0-25

(Enero 2009, Febrero 2008)

14. ¿Para qué sirven las “tablas -o mapas- inversos” de páginas?

- a) Para mantener qué páginas se han usado recientemente
- b) Para reducir el tiempo de búsqueda de qué marcos pertenecen a un proceso
- **c) Para reducir el tiempo de búsqueda de qué procesos usan un marco**
- d) Para la selección de la víctima del TLB

(Enero 2009, Febrero 2008)

15. ¿Cuál de los siguientes se considera como “páginas anónimas” en gestión de memoria?

- a) Código de librerías compartidas
- **b) Asignación dinámica (malloc())**
- c) Código de programas ejecutables
- d) Buffer-cache de disco

(Enero 2009)

4. El sistema RCU usado para reemplazar a lectores-escritores:

- **a) Perjudica a los escritores y beneficia a los lectores**
- b) Beneficia a los escritores y perjudica a los lectores
- c) Es equivalente para ambos
- d) Aumenta el tiempo de latencia de ambos

(Septiembre 2006)

20. ¿Qué llamada de sistema hace uso de la variable de entorno PATH?

- a) execlp
- b) fork
- c) execv
- d) getcwd

(Febrero 2007 M1)

2. La llamada de sistema mmap()

- a) No puede ser usada por procesos de usuario
- b) Facilita las operaciones de ficheros con tamaños variables F, Fuente:
A1-29
- c) Reemplazan completamente a las funciones de E/S basadas en read/
write
- **d) Se basan en el uso de las tablas de páginas de la memoria virtual**