

Apellidos:	Nombre:
Grupo: <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> PCEO	DNI:

Instrucciones para realizar el test:

- La puntuación de este test es de 7 puntos.
- Marca con una «X», en la tabla de respuestas y en **bolígrafo**, una única respuesta para cada cuestión.
- Una respuesta incorrecta resta una respuesta correcta. Una pregunta sin contestar ni suma ni resta.
- Debes entregar la hoja del examen al acabar.**
- Supondremos que 1 KiB = 2¹⁰ bytes, 1 MiB = 2²⁰ bytes y 1 GiB = 2³⁰ bytes. También supondremos que **no existe caché de disco** en aquellos casos en los que haya lecturas y/o escrituras de disco.

Nº	a	b	Respuestas del test	Nº	a	b
1				21		
2				22		
3				23		
4				24		
5				25		
6				26		
7				27		
8				28		
9				29		
10				30		
11				31		
12				32		
13				33		
14				34		
15				35		
16				36		
17				37		
18				38		
19				39		
20				40		

Preguntas:

- Un sistema operativo necesita obligatoriamente el apoyo del hardware para:
 - Impedir que un proceso se apodere de la CPU.
 - Ser multiusuario cuando el propio sistema operativo es ya multitarea.
- Decimos que un sistema operativo se puede ver como una máquina extendida o virtual porque ...
 - Oculto los detalles del hardware real y presenta una nueva interfaz de la máquina que se usa a través de llamadas al sistema.
 - Además de permitir el acceso directo a bajo nivel al hardware de la máquina por parte de los procesos cuando se encuentran en modo usuario, ofrece abstracciones adicionales que el hardware en sí mismo no posee.
- En cuanto a la multitarea es cierto que:
 - Su objetivo es solapar los tiempos de espera de los procesos de un usuario con el uso de la CPU de otros procesos pertenecientes al mismo usuario.
 - Trata de dar la ilusión al usuario de que su programa es el único que se está ejecutando, cuando en realidad la CPU puede estar procesando varios programas de distintos usuarios.
- En la creación de procesos en Unix:
 - Tras la llamada a `fork()`, los procesos padre e hijo tienen diferente PID. La llamada a `exec()` construye la nueva imagen del proceso a partir del ejecutable, sin alterar su PID.
 - Tras la llamada a `fork()`, tanto el proceso padre como el hijo tienen el mismo PID. La llamada a `exec()` es la que le asigna un nuevo PID al proceso que la ejecuta, al tiempo que construye su nueva imagen a partir del ejecutable.
- Con respecto a los estados de los procesos:
 - Se puede pasar directamente de Bloqueado a Saliente.
 - No se puede pasar directamente de Ejecución a Listo Suspendido.
- En cuanto a los hilos:
 - No existe protección entre hilos de distintos procesos ya que la protección sólo se aplica a nivel de proceso, no a nivel de hilo.
 - Los hilos de un mismo proceso se pueden comunicar entre sí sin intervención del sistema operativo.
- En relación con la implementación de los hilos en modo núcleo:
 - Los cambios de contexto entre hilos de un mismo proceso son mucho más rápidos que en una implementación en modo usuario, gracias a la tabla de hilos gestionada por el núcleo.
 - Permite aprovechar el hecho de que haya múltiples CPUs en el sistema, de forma que varios hilos del mismo proceso se podrían ejecutar a la vez.
- Siempre que se cambia de modo usuario a modo núcleo ...
 - Es porque se ha producido una interrupción, excepción o llamada al sistema.
 - Se hace dentro de un cambio de proceso, ya que un cambio de modo solamente no tiene sentido.
- Cuando un proceso bloqueado se desbloquea, continúa su ejecución:
 - Inmediatamente, para no perjudicar su interactividad con el usuario.
 - Cuando lo decida el algoritmo de planificación utilizado.
- La planificación Round Robin (RR):
 - Puede hacer que un proceso que agota su quantum tenga que esperar hasta $(n + 1) \cdot q$ unidades de tiempo (n =nº de procesos, q =quantum) para volver a ejecutarse en la CPU.
 - Es equitativa entre todos los procesos

11. Supongamos un proceso por lotes con las siguientes ráfagas:

CPU	E/S	CPU
4	3	3

Si el proceso llega y realiza su primer tic de ejecución en el instante 11, y realiza el último tic de ejecución y acaba en el 22, entonces:

- a) Su tiempo de retorno es 13 y su tiempo de espera 2.
- b) Su tiempo de retorno es 12 y su tiempo de espera 2.

12. Tenemos un SO que utiliza el algoritmo de planificación de procesos SRTF con oráculo. Esta es la situación justo cuando va a comenzar la u.t. 5:

- En la CPU: Proceso P0 ejecutando una ráfaga de 6 u.t. que comenzó en la u.t. 3.
- En la cola de listos y en este orden de llegada:
 - Proceso P1 para ejecutar una ráfaga de 6 u.t.
 - Proceso P2 para ejecutar una ráfaga de 3 u.t.
 - Proceso P3 para ejecutar una ráfaga de 5 u.t.

Entonces es cierto que:

- a) El proceso P0 acabará de ejecutarse justo antes de comenzar la u.t. 12.
- b) El proceso P0 acabará de ejecutarse justo antes de comenzar la u.t. 17.

13. Comparando las listas de control de acceso (ACL) y las listas de capacidades:

- a) La capacidad puede ser copiada, ya que se comporta como una especie de ticket que te da acceso al recurso, mientras que las ACL no ofrecen esa posibilidad.
- b) Las ACL permiten un acceso muy rápido a la lista de recursos a la que tiene acceso un usuario dado, mientras que las listas de capacidades permiten un acceso muy rápido a la lista de usuarios que tienen acceso a un recurso determinado.

14. En Linux:

- a) La ejecución de un programa con SETUID/SETGID puede implicar un cambio de dominio.
- b) Los bits SETUID/SETGID modifican el UID y el GID con que se ejecuta el programa, aunque el usuario y el grupo efectivo sigue siendo el mismo.

15. En UNIX se utiliza una base aleatoria junto a la contraseña proporcionada por el usuario. El motivo es ...

- a) Histórico, pero actualmente inútil porque esa base se guarda junto con la clave pero sin encriptar, así que cualquiera puede conocerla.
- b) Evitar que para dos usuarios con la misma contraseña se acabe generando el mismo valor encriptado.

16. Si un directorio de Unix tiene los permisos `usuario1 grupo1 rwxr-xr-x`, entonces, sin tener en cuenta al usuario `root`:

- a) Sólo el `usuario1` podrá escribir en los ficheros del directorio.
- b) Sólo el `usuario1` podrá eliminar ficheros del directorio.

17. De entre las dos alternativas siguientes para la implementación de ficheros, ¿cuál daría lugar a fragmentación externa?

- a) Asignación mediante lista ligada.
- b) Asignación adyacente.

18. Si en una implementación de ficheros mediante nodos-i y bloques indirectos, los bloques lógicos son de 1 KiB y las direcciones de bloque ocupan 4 bytes, entonces, a través de cada bloque doblemente indirecto, usando los correspondientes bloques simplemente indirectos, se podrá llegar a:

- a) 65536 bloques de datos.
- b) 1024 bloques de datos.

19. En un sistema de ficheros Unix, el número máximo de ficheros que se pueden crear:

- a) Depende exclusivamente del número de nodos-i de que dispone.
- b) Depende exclusivamente del número de bloques lógicos que haya en la zona de datos.

20. El objetivo de una caché de disco como la vista en clase es ...

- a) Mejorar el rendimiento de las operaciones de lectura de disco. Las escrituras de disco siempre se hacen directamente en el disco para evitar que se pierda la consistencia ante una caída del sistema.
- b) Mejorar el rendimiento de las operaciones de lectura/escritura de los discos.

21. Suponiendo que tenemos un sistema de ficheros UNIX en el que el nodo-i del directorio raíz NO se encuentra en memoria aún, que cada directorio ocupa un bloque de disco y que cada nodo-i que busquemos se encuentra en un bloque diferente, ¿cuántos bloques habrá que leer para cargar en memoria el nodo-i del fichero cuya ruta es `/home/invitado/fichero.txt`?

- a) 7
- b) 6

22. Si un disco, con una única partición, usa sectores de 512 bytes y el sistema de ficheros almacenado en esa partición usa bloques lógicos de 4096 bytes, entonces, un fichero que almacena 510 bytes de datos a partir de su posición 0, ocupará:

- a) Exactamente un bloque lógico.
- b) Exactamente un sector.

23. En Unix, si creamos un fichero *F* y, a continuación, creamos 5 enlaces a *F*, de los que 2 son simbólicos y 3 son físicos, ¿qué valor tendrá el campo de número de enlaces en el nodo-i de *F*?

- a) 4
- b) 6

24. Las entradas de un directorio se leen con `readdir` y no con `read` porque:

- a) Los directorios almacenan sus contenidos en *ficheros de directorio*, que se implementan de forma diferente al resto de ficheros y, por tanto, no pueden ser accedidos con una operación `read`.
- b) `readdir` permite aislar a los procesos de los detalles de la implementación de los directorios; esto hace que un proceso pueda seguir accediendo al contenido de un directorio aun cuando su implementación cambie.

25. A la hora de resolver el problema de la protección entre procesos que surge en los esquemas de administración de memoria con particiones:

- a) El hardware debe proveer necesariamente algún mecanismo, por ejemplo, registros especiales, y realizar el chequeo de las direcciones de memoria generadas por los procesos.
- b) No se requiere ningún soporte por parte del hardware, ya que es el sistema operativo el que irá comprobando las direcciones de memoria, y si alguna está fuera del rango permitido, abortará al proceso correspondiente.

26. Si comparamos la segmentación paginada con la segmentación pura, podemos ver que:

- a) En la segmentación pura podemos tener fragmentación externa, y en la segmentación paginada, no.
- b) En ambas, para cada segmento, tenemos una base con la dirección de inicio del segmento en memoria principal y un límite que nos determina la longitud o tamaño del mismo.

27. El bit de «Caching Desactivado» que aparece en cada entrada de la tabla de páginas, sirve para:

- a) Implementar la política de «Escritura Anticipada», de forma que se escriban en disco aquellas páginas modificadas en cuya entrada dicho bit toma valor 1.
- b) Evitar que el contenido de ciertas páginas (aquellas en los que dicho bit está a 1) se almacene en la memoria caché del procesador.

28. El «algoritmo de frecuencia de fallos de página»:
- Es un algoritmo utilizado para el reparto de marcos entre procesos.
 - Es un algoritmo de reemplazo de páginas.
29. En las técnicas de memoria virtual estudiadas, existe siempre el siguiente número de espacios de direcciones independientes:
- En paginación y en segmentación paginada 1, en segmentación 3 (texto, datos, pila).
 - En paginación 1, en segmentación y en segmentación paginada tantos como segmentos existan.
30. El mapa de memoria de un proceso es una estructura de datos que:
- Describe cómo está organizado el espacio de direcciones virtuales de un proceso (regiones que lo conforman, características de las mismas, etc).
 - Registra qué zonas de memoria RAM están asignadas a un proceso (número de zonas, tamaño y dirección de cada una, etc).
31. Dada una tabla de páginas lineal y una tabla de páginas invertida, es cierto que:
- El tamaño de cada entrada en la tabla de páginas invertida suele ser mayor que el tamaño de cada entrada en la tabla de páginas lineal.
 - El número total de entradas en la tabla de páginas lineal suele ser menor que el número total de entradas en la tabla de traducción de la tabla de páginas invertida.
32. El mecanismo de «copia en escritura»:
- Mejora el rendimiento copiando a un marco físico de reserva una página virtual sucia que debe ser escrita, en lugar de llevarla a disco.
 - Facilita, por motivos de eficiencia, que un proceso padre y su hijo accedan a la misma página de memoria hasta que uno de ellos la modifique. En ese momento se duplica la página modificada para evitar que el otro proceso vea el cambio.
33. Si tenemos particionado estático de memoria:
- Si hay dos particiones menores libres, que unidas permitan que el proceso quepa, se podrán juntar y ejecutarlo en ellas.
 - No se podrá ejecutar un proceso si no hay ninguna partición disponible con un tamaño mayor o igual que el del proceso.
34. Supongamos una tabla de páginas multinivel con N niveles. Para un acceso a memoria a la página virtual V , que falla en el TLB pero que acierta en la tabla de páginas, el número de accesos a memoria principal requeridos para completar la traducción es:
- N , en todos los casos.
 - Desde 1 hasta N , dependiendo del nivel de la tabla de páginas en el que se encuentre la entrada asociada a V .
35. Supongamos un sistema operativo que utiliza una política de asignación estática y un proceso que tiene asignados 3 marcos de página. Si el proceso ha realizado los siguientes accesos (en este orden) WP_1 , WP_2 , RP_3 , donde R y W son operaciones de lectura y escritura respectivamente, y P_i son i páginas distintas de memoria, un nuevo acceso RP_4 ocasiona un fallo de página que supone el reemplazo de:
- La página P_1 si se utiliza un algoritmo de reemplazo FIFO segunda oportunidad y la página P_3 si se utiliza NRU.
 - La página P_1 si se utiliza el algoritmo de maduración y la página P_3 si se usa LRU.
36. En el algoritmo de planificación de disco SSF:
- Siempre se consigue un tiempo promedio óptimo cuando el dispositivo de almacenamiento es un disco duro.
 - Una solicitud corre el riesgo de no ser atendida durante mucho tiempo en un disco muy cargado.
37. Un sistema de *spooling* es útil en un sistema multiprogramado para la gestión de ...
- Un disco duro.
 - Una impresora.
38. El manejador del reloj se encarga directamente, entre otras tareas, de:
- Mantener la hora, registrar el uso de la CPU y activar el motor de la unidad de DVD.
 - Evitar la monopolización de la CPU, registrar el uso de la CPU y controlar las alarmas.
39. Tenemos un ordenador que trabaja a M MIPS, cuyo reloj dispone de un oscilador de cuarzo a F MHz y de un registro de carga con el valor C . El número máximo de instrucciones que puede tener el manejador del reloj para poder tratar correctamente todas las marcas de reloj que se produzcan es:
- $\frac{M \times C}{F}$
 - $\frac{F \times M}{C}$
40. Dado un disco girando a R RPM, con S sectores por pista (todas las pistas de igual tamaño), el tiempo medio de latencia se calcula como:
- $\frac{60}{S \times R}$ seg.
 - $\frac{30}{R}$ seg.

Nº	a	b
1	X	
2	X	
3		X
4	X	
5	X	
6		X
7		X
8	X	
9		X
10		X
11		X
12	X	
13	X	
14	X	
15		X
16		X
17		X
18	X	
19	X	
20		X

Respuestas del test

Nº	a	b
21	X	
22	X	
23	X	
24		X
25	X	
26	X	
27		X
28	X	
29		X
30	X	
31	X	
32		X
33		X
34	X	
35	X	
36		X
37		X
38		X
39	X	
40		X