

Apellidos:	Nombre:
DNI:	Grupo y profesor:

Instrucciones para realizar el test:

- La puntuación de este test es de 10 puntos.
- Tiempo recomendado: **30 minutos**.
- No olvides poner tus apellidos, nombre, DNI y grupo al que perteneces al principio de la hoja.
- 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.** El test no puede sacarse del aula.
- Supondremos que $1 \text{ KiB} = 2^{10} \text{ bytes}$, $1 \text{ MiB} = 2^{20} \text{ bytes}$ y $1 \text{ GiB} = 2^{30} \text{ bytes}$. 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				16		
2				17		
3				18		
4				19		
5				20		
6				21		
7				22		
8				23		
9				24		
10				25		
11				26		
12				27		
13				28		
14				29		
15				30		

Preguntas:

- A la hora de implementar las llamadas al sistema:
 - Suele existir un mecanismo hardware distinto para implementar cada categoría de llamadas al sistema, ya que puede haber grandes diferencias entre unos tipos de llamadas al sistema y otros.
 - Todas suelen usar el mismo mecanismo proporcionado por el hardware a través del cual el procesador cambia de modo usuario a modo núcleo de forma controlada.
- Las llamadas al sistema:
 - son la única forma de pedirle un servicio al núcleo del sistema operativo.
 - es imposible hacerlas directamente, de ahí que se hagan a través de una biblioteca del sistema.
- El intérprete de órdenes es un programa del sistema que funciona:
 - en modo usuario, haciendo uso de las llamadas al sistema que proporciona el sistema operativo.
 - dentro del núcleo del sistema operativo ya que necesita hacer operaciones que cualquier proceso de usuario no puede.
- De los mecanismos de protección hardware que ayudan al sistema operativo a proteger el sistema:
 - No se puede prescindir de ninguno de los tres.
 - Las interrupciones periódicas no son necesarias realmente, ya que no protegen de nada, mientras que el modo dual evita que se ejecuten instrucciones peligrosas y la protección de memoria evita que un proceso pueda salirse de la memoria asignada.
- Una ventaja de los sistemas operativos que tienen una estructura cliente-servidor respecto a los monolíticos es que:
 - Se pueden ejecutar sobre un mayor número de arquitecturas de ordenadores.
 - Son más fáciles de depurar y verificar.
- ¿Por qué un sistema operativo se puede ver como una máquina extendida o virtual?
 - Porque oculta los detalles del hardware real y presenta una nueva interfaz de la máquina que se usa a través de llamadas al sistema.
 - Porque, además de permitir el acceso 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.
- Un proceso suspendido-listo puede pasar a uno de los siguientes estados:
 - A listo cuando es reanudado.
 - A suspendido-bloqueado cuando empieza a servirse su petición de E/S.
- El efecto convoy ...
 - puede darse en un sistema que planifica el procesador con el algoritmo SJF.
 - aparece en FCFS cuando varios procesos limitados por E/S tienen que estar esperando a que deje la CPU un proceso limitado por CPU.
- Siempre que se cambia de modo usuario a modo núcleo ...
 - Se hace dentro de un cambio de proceso, ya que un cambio de modo solamente no tiene sentido.
 - Es porque se ha producido una interrupción, excepción o llamada al sistema.
- En Unix, un proceso que se desbloquea se despierta en ...
 - el núcleo del SO, tras la función/instrucción que hizo que se bloqueara en última instancia.
 - la instrucción en el código de usuario detrás de la llamada al sistema que provocó que fuera bloqueado.
- En el algoritmo de planificación de procesos de múltiples colas con realimentación, la realimentación significa:
 - Que un proceso puede pasar de una cola a otra.
 - Que un proceso puede pasar de una cola a otra y que además se puede cambiar la posición de un proceso en su cola en cualquier momento.

12. Indica cuál de los siguientes es un motivo por el que se lanza obligatoriamente el planificador de procesos.
- Se ejecuta una llamada al sistema.
 - Finaliza un proceso.

13. En la creación de procesos en Unix:
- 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.
 - 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.

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

CPU	E/S	CPU
4	3	3

Si el proceso llega en el instante 11 y termina en el 23 entonces:

- Su tiempo de retorno es 13 y su tiempo de espera 2.
 - Su tiempo de retorno es 12 y su tiempo de espera 2.
15. Supongamos que en la cola de procesos listos tenemos tres procesos, P_1 , P_2 y P_3 , en ese orden, cuya siguiente ráfaga de CPU durará 4, 3 y 5 unidades de tiempo, respectivamente. Si antes de que la CPU quede libre, llega a la cola de listos un proceso P_4 cuya siguiente ráfaga de CPU va a durar 2 unidades de tiempo, entonces, dicho proceso tomará la CPU:
- Como mucho, 8 unidades de tiempo después de llegar, si se usa round-robin con un quantum de 2 unidades y la CPU acaba de ser asignada a un proceso.
 - Inmediatamente, expulsando al proceso que hay en la CPU, si el algoritmo es SRTF o por prioridad apropiativo (siempre que P_4 sea el proceso de mayor prioridad).
16. Un *rootkit* es un programa que permite un acceso continuo y privilegiado a un computador y, además:
- Tiene la capacidad de extenderse a otros computadores conectados a su misma red.
 - Es difícilmente detectable, incluso mediante un antivirus.
17. Una característica de las listas de posibilidades que no tienen las listas de control de acceso es:
- que se permiten operaciones con las posibilidades, como obtener una copia o eliminarlas.
 - que es más sencillo recuperar todos los permisos de un objeto particular.

18. Un dominio de seguridad:
- Es un conjunto de parejas (objeto, derechos)
 - Especifica los permisos que posee un usuario

19. Si tenemos un fichero con los siguientes permisos: `usuario1 grupo2 rwsr-sr--`:
- si lo ejecuta el `usuario1` que pertenece al `grupo1`, su grupo se convierte en el `grupo2` durante la ejecución del proceso.
 - no puede ejecutarlo el `usuario2` que pertenece al `grupo2` porque no tiene permiso de ejecución (x) para el grupo.

20. En UNIX/Linux:
- Un usuario puede pertenecer a varios dominios a la vez con distintos GID.
 - Un usuario puede pertenecer a varios dominios a la vez con distintos UID y GID.

21. Los privilegios de un usuario en un sistema:
- dependen tanto de su identificador como de su contraseña.
 - dependen únicamente de su identificador, ya que la contraseña solamente sirve para autenticarlo.

22. En la implementación de ficheros y directorios en Unix, los bloques indirectos:
- Los usan tanto los ficheros regulares como los directorios (si es necesario).
 - Los usan los ficheros regulares (si es necesario), pero no los directorios.

23. En la implementación de ficheros mediante lista ligada:
- Se consiguen utilizar todos los bloques del disco.
 - Se produce mucha fragmentación externa.

24. Un fichero ejecutable que contiene código máquina es un fichero binario cuyo formato debe conocer:
- Sólo el compilador-enlazador que genera dicho fichero binario. En los sistemas operativos modernos, el tipo de los ficheros es indiferente.
 - Tanto el compilador-enlazador que lo genera como el sistema operativo donde se utiliza.

25. El campo «enlaces» de un nodo-i:
- guarda el número de subdirectorios en el caso del nodo-i del directorio raíz.
 - guarda el número de entradas en directorios con ese nodo-i.

26. ¿En qué implementación de directorios, de las vistas en clase, pueden aparecer huecos libres entre entradas de directorio?
- Tanto en la implementación de directorios de Unix como en la de MS-DOS.
 - Sólo en la implementación de directorios de Unix.

27. Las entradas de un directorio se leen con `readdir` y no con `read` porque:
- Los directorios no se implementan como ficheros.
 - `readdir` permite aislar a un proceso de los detalles de la implementación del directorio.

28. El número máximo de bloques lógicos que se deben leer de un disco para copiar en memoria el bloque N de un fichero es:
- 1 en asignación contigua.
 - $N - 2$ con listas ligadas.

29. Si queremos escribir un byte en la posición 4 500 000 (contando desde 0) de un fichero de 10 MB implementado mediante lista ligada sin índice, utilizando bloques lógicos de 4 kB y direcciones de disco de 4 bytes, entonces tendremos que:
- Leer 1100 bloques y escribir 2 bloques
 - Leer 1100 bloques y escribir 1 bloque.

30. Si queremos añadir un bloque al final de un fichero con un tamaño de 10 MB implementado mediante nodos-i capaces de almacenar 13 direcciones (10 directas, 1 BSI, 1 BDI y 1 BTI), utilizando bloques lógicos de 4 kB, direcciones de disco de 4 bytes y suponiendo el nodo-i ya en memoria, entonces tendremos que:
- Leer 2 bloques y escribir 1 bloque.
 - Leer 2 bloques y escribir 2 bloques.

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	

Respuestas del test

Nº	a	b
16		X
17	X	
18	X	
19	X	
20	X	
21		X
22	X	
23	X	
24		X
25		X
26	X	
27		X
28	X	
29		X
30		X