

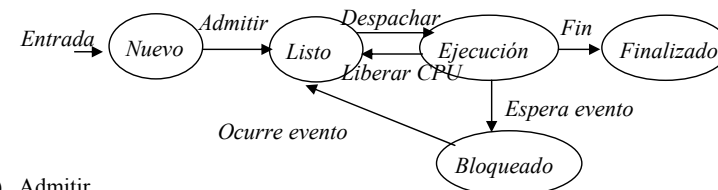
Escuela Universitaria de Ingeniería Técnica en Informática de Oviedo Sistemas Operativos. Segundo curso. Examen Final Junio. Primer Parcial. 23 de Febrero de 2002	Modelo 1
---	-----------------

Puntuación: Acierto: + 3 Fallo: -1 En Blanco: 0 Puntuación mínima para aprobar: 34 puntos Nota sobre 10 = (puntos - 34)/26 * 5 + 5 Tiempo de realización del examen: 45 minutos
--

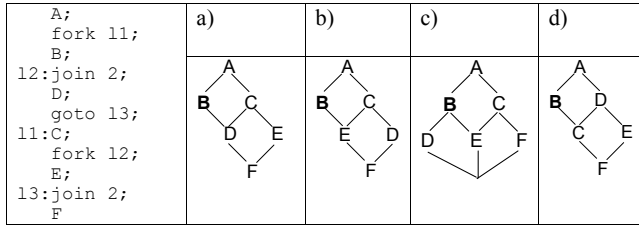
- 1.- ¿Cuál de las siguientes es una ventaja de un sistema estructurado por capas frente a uno monolítico?
 - a) Permiten incorporar en modo usuario funciones del núcleo.
 - b) Su modularidad facilita su implementación y depuración.
 - c) Ofrecen un mejor rendimiento.
 - d) Posibilitan incorporar un mayor número de funciones.
- 2.- Al introducirse el tiempo compartido, se introdujo una nueva causa por la que un proceso puede abandonar la CPU con respecto a la multiprogramación existente hasta entonces. Señale cuál.
 - a) La realización de una e/s por parte del proceso.
 - b) La terminación del proceso
 - c) La expiración del cuanto.
 - d) Todas las señaladas son causas que ya existían en la multiprogramación.
- 3.- ¿Qué ventaja aporta la introducción de la multiprogramación?
 - a) Que el operador tiene que trabajar menos.
 - b) Que los usuarios pueden interactuar con el sistema directamente.
 - c) Que el sistema está más tiempo ocupado.
 - d) Todas las señaladas son ventajas que introdujo la multiprogramación.
- 4.- ¿La aparición de qué tipo de sistema operativo cambió la forma de interactuar del usuario final con el sistema?
 - a) Los sistemas de procesamiento por lotes.
 - b) Los sistemas multiusuario.
 - c) Los sistemas interactivos.
 - d) Los sistemas con multiprogramación.
- 5.- Cuando a un proceso en Unix se le acaba el cuanto, pasa al estado
 - a) Dormido.
 - b) Preparado para ejecución en memoria principal.
 - c) Requisado.
 - d) Preparado para ejecución en memoria secundaria.
- 6.- ¿Dónde se almacena el vector de interrupciones?
 - a) Hay una única copia al comienzo de la tabla de procesos del sistema.
 - b) Hay una copia en el PCB de cada proceso.
 - c) En una zona de memoria determinada por el hardware del sistema.
 - d) En una zona de memoria determinada por el sistema operativo.
- 7.- ¿Qué parte del sistema operativo es la encargada de llevar a cabo el cambio de contexto?
 - a) El planificador a medio plazo.
 - b) El planificador a corto plazo.
 - c) El despachador.
 - d) El planificador a largo plazo.
- 8.- De las siguientes, una no es una causa que justifique la necesidad de contar con un administrador del sistema operativo. Señálela.
 - a) Incremento en la complejidad de los sistemas.

Sistemas Operativos	Modelo 1	Página 2
---------------------	----------	----------

- b) Resolver las dudas de programación de los usuarios.
 - c) Todas las señaladas son causas para tener un administrador.
 - d) Aumentar la productividad del sistema.
- 9.- De los siguientes mecanismos de sincronización, sólo uno suele ser gestionado por el sistema operativo. Señale cuál.
 - a) Regiones Críticas.
 - b) Semáforos.
 - c) Monitores.
 - d) Regiones Críticas condicionadas.
- 10.- La llamada al sistema *signal* en Unix se utiliza para
 - a) Hacer que el proceso que la ejecuta se duerma.
 - b) Definir el comportamiento del proceso ante una señal.
 - c) Enviar señales a procesos.
 - d) Ninguna del resto de las respuestas es correcta.
- 11.- El formato de los mensajes en Unix incluye un campo “tipo”. ¿Cuál es su utilidad?
 - a) Ninguna. Es meramente informativo.
 - b) Sirve para indicar el tipo (carácter, entero, real, etc) de los datos que contiene el mensaje.
 - c) Se utiliza para poder recibir mensajes de un tipo determinado, modificando el comportamiento FIFO normal de la cola.
 - d) Ninguna del resto de respuestas es correcta.
- 12.- El intento de solución al problema de la sección crítica basado en el uso de la variable “turno” no es válido porque:
 - a) No garantiza la exclusión mutua.
 - b) No tiene ningún problema. Es una solución válida.
 - c) Fuerza el uso alternativo del recurso.
 - d) Puede producir interbloqueo.
- 13.- En el modelo de ciclo de vida siguiente, ¿sobre qué transiciones se ejecuta el Planificador a largo plazo?



- a) Admitir
 - b) Admitir y Fin
 - c) Ninguna de las anteriores es correcta o más de una lo es
 - d) Admitir, Despachar y Liberar CPU
- 14.- ¿Cuál de las siguientes políticas de planificación del procesador elimina la posibilidad de inanición de procesos intensivos en CPU?
 - a) Ninguna es correcta
 - b) SJF Primero el más corto
 - c) Round Robin + FIFO
 - d) SRT Menor tiempo restante
- 15.- Señale cuál de los siguientes grafos de precedencia corresponden con el siguiente código concurrente:



- 16.- ¿Cuál de las siguientes transiciones no se puede producir en un sistema con un algoritmo de planificación a corto plazo no apropiativo?
- Ninguna es correcta
 - de bloqueado a listo
 - de ejecución a bloqueado
 - de ejecución a listo
- 17.- Un sistema de gestión de memoria basado en particiones variables mantiene la siguiente lista de bloques libres en el momento actual ([dirección inicial del hueco, tamaño del hueco]):
 Lista de Huecos = [12K, 26K], [47K, 50K], [138K, 40K], [256K, 128K], [500K, 71K], [602K, 73K].
 La lista está ordenada siempre por direcciones crecientes de inicio de hueco, de forma que cualquier nuevo hueco que se genere se insertará en la posición que le corresponda. Se desea ejecutar los procesos P1, P2 y P3, que llegan e intentan cargarse en memoria en este orden y con el tamaño siguiente: P1→50K, P2→20K y P3→75K. ¿cuál sería la respuesta correcta?
- Ninguna
 - Utilizando el primer ajuste, la lista de huecos tras la carga de los tres procesos sería: ListaHuecos→ [32K, 6K], [138K, 40K], [331K, 53K], [500K, 71K], [602K, 73K]
 - Utilizando el peor ajuste, la lista de huecos tras la carga de los tres procesos sería: ListaHuecos→ [12K, 26K], [47K, 50K], [138K, 40K], [331K, 53K], [520K, 51K], [652K, 23K]
 - Utilizando el mejor ajuste, la lista de huecos tras la carga de los tres procesos sería: ListaHuecos→[32K, 6K], [138K, 40K], [256K, 128K], [500K, 71K]
- 18.- Supóngase que se quiere implantar una rutina *debug()* que recorre la información del proceso en ejecución e imprime una serie de datos como el tamaño, el conjunto de direcciones físicas de memoria ocupadas por el texto, los datos y la pila del proceso, la prioridad y el valor actual del contador de programa. ¿A qué estructura de datos se necesita acceder?
- Ninguna de las anteriores
 - Es necesario acceder a la memoria ocupada por texto, datos y pila del proceso para averiguar dónde comienza y dónde acaba el proceso
 - Todo se puede obtener del PCB
 - Todo se obtiene de los registros

19.- Un sistema de gestión de memoria basado en particiones fijas divide la memoria principal de la siguiente forma:

12 K	Partición 0
32 K	Partición 1 (libre)
64 K	Partición 2 (libre)
128 K	Partición 3 (libre)

Se pretende ejecutar un programa P de tamaño 30K. Suponiendo una política de asignación del tipo mejor ajuste, cárguese el programa P en la partición correspondiente. Si durante su ejecución, P generase la dirección lógica 24827 (todas las direcciones direccionan bytes de memoria principal), ¿cuál sería la dirección física correspondiente?

- Se produciría una excepción de dirección inválida
- 37115
- Ninguna de las anteriores
- 69883

20.- En un sistema paginado con tamaño de página 1K, sean dos procesos P y Q de tamaño 5000 bytes y 4000 bytes respectivamente, con las siguientes tablas de página:

P		
Nº página	Otra información	Nº marco
0	7
1	23
2	45
3	82
4	36
5	44
...	65

Q		
Nº página	Otra información	Nº marco
0	23
1	34
2	33
3	46
4	7
5	44
...	41

- Las direcciones lógicas que P y Q utilizan para acceder al marco 7 son las mismas
- P y Q comparten los marcos de página 7, 23 y 44
- P y Q comparten el marco de página 23
- Ninguna de las anteriores