

Examen Tema 4. Fundamentos de los Sistemas Operativos

1º DAM. Sistemas informáticos. IES Valle Inclán. Curso 2020/2021

1. Responde

- 1.1. ¿En un sistema monotarea, cuántos procesos diferentes al SO están a la vez en la memoria?

Uno sólo, por eso se llama MONO-tarea.

- 1.2. ¿Qué diferencia existe entre un programa y un proceso?

Que un proceso es un programa en ejecución.

- 1.3. ¿Qué número identifica de forma única a un proceso respecto al SO?
Indica la sigla y su significado

El PID (process identification). Se trata de un número entero usado por el Kernel para identificar a los diferentes procesos de forma unívoca. Para ello, utiliza una variable global que va incrementando el nº PID con cada nuevo proceso que se crea mediante una llamada fork().

2. Indica en la siguiente tabla la evolución histórica de los SO escribiendo las generaciones (1ª, ...) y especifica en cada generación los hechos más importantes que marcaron cada una tanto a nivel de hardware como a nivel los conceptos que se introdujeron en los SO

Generación	A nivel hardware	A nivel SO
1ª Generación (1945-1955)	Los ordenadores estaban hechos con los llamados tubos de vacío. A finales de los 50 se introducen los paneles enchufables	No había SO. Se programaba directamente sobre el hardware en lenguaje máquina.
2ª Generación (1955-1965)	Aparecen los transistores, logrando fabricar ordenadores más pequeños y más potentes	Se comienza a programar en lenguaje ensamblador sobre tarjetas perforadoras y se hacía en FORTRAN. parece el procesamiento por lotes, y posteriormente éste se mejora al procesamiento fuera de línea (off-line) conjunto con la invención de las cintas magnéticas y las impresoras en línea
3ª Generación (1965-1980)	Se crean los circuitos integrados (CI)	Desarrollo de la multiprogramación
4ª Generación (1980-hoy día)	Se crean los circuitos integrados a gran escala (LSCI)	Aparecen los primeros sistemas operativos como MS-DOS y MacOS.

		La tendencia es evolucionar hacia GUIs (interfaces gráficas) más amigables e intuitivas para el manejo del usuario, ya que comenzó el auge de los ordenadores personales.
--	--	---

3. Escribe en la siguiente tabla, la agrupación de los SO en función de una serie de criterios, es decir, por ejemplo en base a su estructura los SO se pueden clasificar en ...

ors

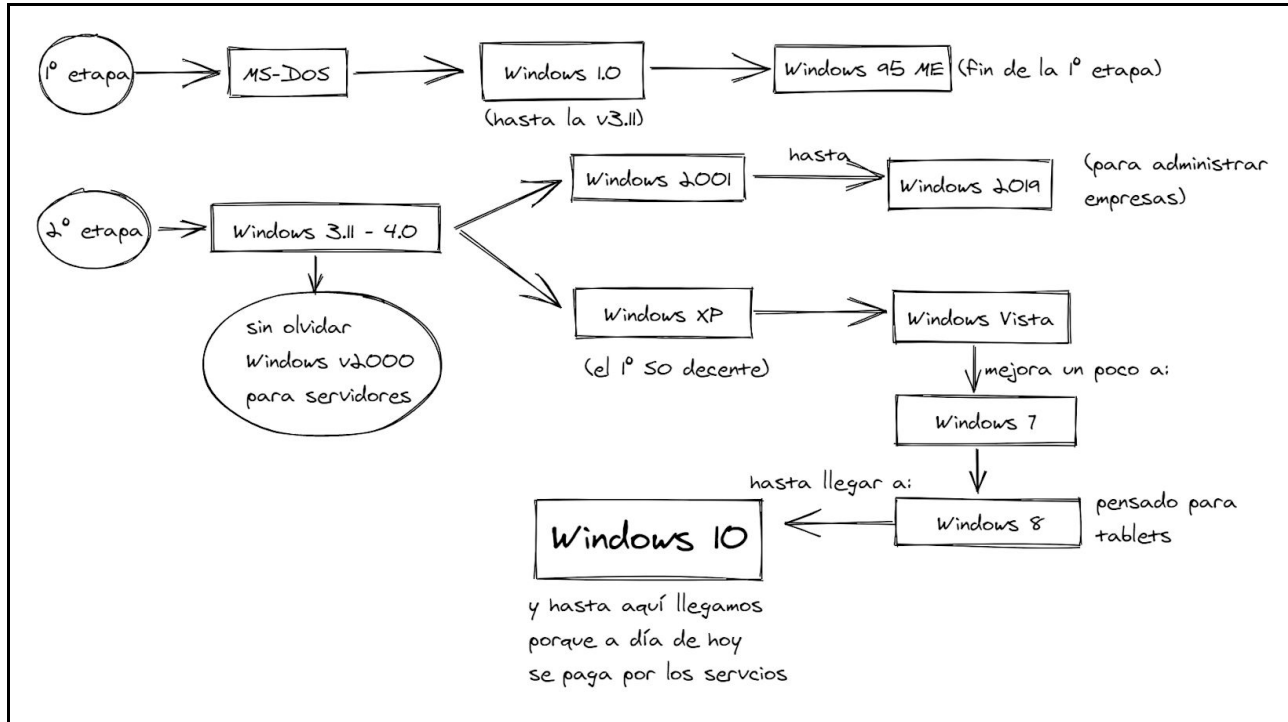
Criterio 1	Criterio 2	Criterio 3
en función de su estructura	en función de los servicios que ofrecen	en función de cómo ofrecen sus servicios
<ul style="list-style-type: none"> - Monolíticos - Jerárquicos - Máquina Virtual 	Según nº de usuarios: <ul style="list-style-type: none"> - monousuario - multiusuario Según nº de tareas: <ul style="list-style-type: none"> - monotarea - multitarea Según nº de procesos <ul style="list-style-type: none"> - Uniproceto - Multiproceto 	<ul style="list-style-type: none"> - SO distribuidos - SO en red

4. Historia y evolución del SO Windows.

A través de la herramienta [excalidraw](#) realiza un dibujo en modo diagrama, de la evolución de Windows, desde ms-dos ⇒ hasta la actualidad.

(realiza tantas ramas evolutivas, como SO distintos se puedan considerar)

OS



5. Según su estructura los SO se pueden estructurar en Monolíticos, en anillos/capas o cliente/servidor (entre otros).

Indica en la siguiente tabla, las ventajas e inconvenientes de los SO en ese tipo de estructura.

	Ventajas	Inconvenientes
Monolíticos		
Anillos/Capas		
Cliente-Servidor		

6. Responde

- 6.1. ¿Cuando hablamos de que un proceso muere de inanición, o se bloquea por este motivo, a qué nos estamos refiriendo?

La inanición (starvation en inglés) ocurre cuando a un proceso o hilo no se le llega a conceder nunca el acceso a los recursos compartidos.
La inanición no es sinónimo de interbloqueo, pero un interbloqueo si puede provocar la inanición de los procesos involucrados.
La inanición puede acabar (aunque no tiene por qué), mientras que un interbloqueo necesita una acción del exterior para finalizar.

- 6.2. ¿Cómo se llama el componente del sistema operativo encargado de la gestión de los procesos?

El planificador de procesos

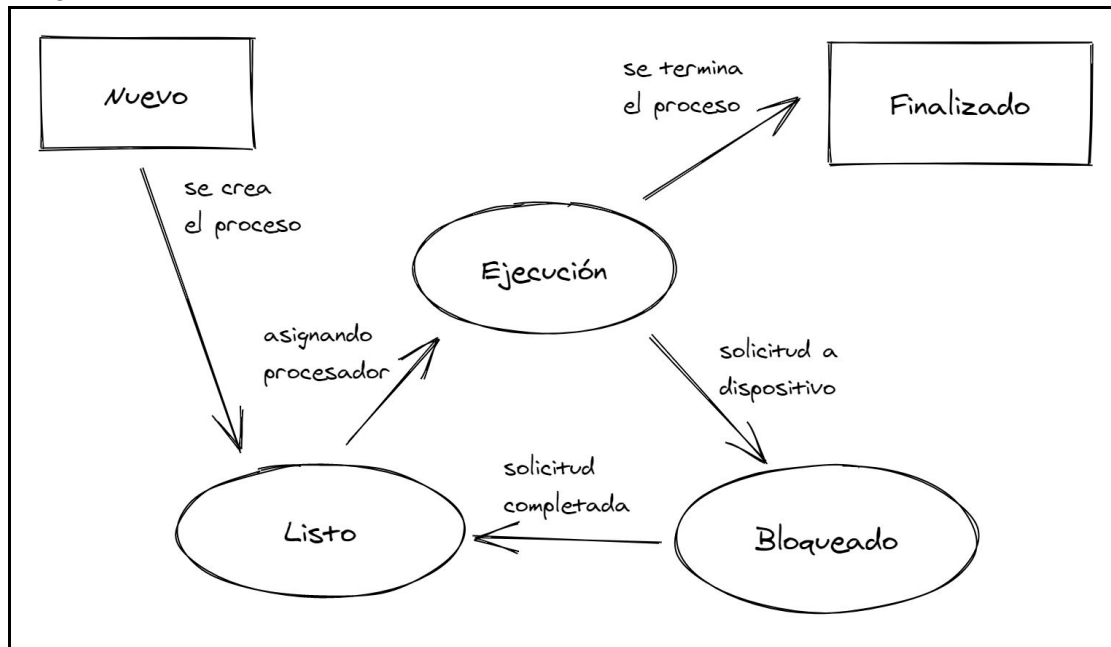
- 6.3. ¿Este componente del SO es en sí mismo un proceso más, que se rige por las mismas reglas que los demás procesos? Razona tu respuesta

NO.
El planificador de procesos es un proceso que se crea inmediatamente al iniciar el sistema operativo. Éste coge todo el uso de la memoria, para luego ir administrando a los demás procesos que la soliciten en función de las tareas que esté llevando a cabo el usuario.
Nota: el PID del planificador de procesos es el 0.

7. Indica y explica cada uno de los estados en los que un proceso puede estar, desde el inicio hasta el fin, y dibuja un diagrama donde se indiquen esos estados y el motivo de porque cambia de estado.

Estados	Descripción
Listo (en espera)	Los procesos en estado listo son los que pueden pasar a estado de ejecución si el planificador del sistema operativo los selecciona, esto es, cuando llegue su turno (según el orden de llegada o prioridad).
En ejecución	Los procesos en estado de ejecución son los que se están ejecutando en el procesador en un momento dado.
Bloqueado	Los procesos que se encuentran en estado bloqueado están esperando la respuesta de algún otro proceso para poder continuar con su ejecución, por ejemplo, una operación de entrada/salida.

Diagrama de estados



8. Responde

04

- 8.1. Cuando se produce un cambio de contexto, el SO debe guardar una serie de información para poder volver a poner de nuevo el proceso en el mismo estado que antes de salir de ejecución, como si no se hubiera producido el cambio de contexto.

¿Qué nombre recibe ese grupo de información?

Bloque de Control *Proceso*

- 8.2. ¿Qué datos o información se guardan en ese grupo de información?

El BC almacena la siguiente información:

- Estado actual del proceso: Ejecución, preparado o bloqueado.
- Identificador del proceso (PID): Dependiendo del SO, a cada proceso se le asigna un PID.
- Prioridad: La asignada por el planificador.
- Ubicación en memoria: Dirección de memoria en la que se carga el proceso.
- Recursos utilizados: Recursos hardware y software para poder ejecutarse.

- 8.3. Para poder controlar toda la información de los procesos, el SO debe disponer de una estructura que permita llevar el control de todos los procesos en ejecución, y la información de cada uno de ellos.

¿Cómo se llama esta estructura y qué dato de un proceso se utiliza para poder localizar a un proceso dentro de esa estructura?

Estructura de Control

Tabla Control Proceso

9. Responde

- 9.1. ¿Cuando el planificador de proceso se ejecuta en modo "Planificación apropiativa" puede expulsar a los procesos del procesador en base a un algoritmo de planificación?

0'45
Exactamente, al ser apropiativa el planificador de procesos puede arrebatarse el uso de la memoria a un proceso determinado. Por otro lado, en la planificación No apropiativa, esto no puede ocurrir.

- 9.2. ¿Qué diferencia existe entre un proceso y una hebra?

Que una hebra es un punto determinado de la ejecución de un proceso, es decir, es un "trozo", una "porción" de proceso en cuestión.

- 9.3. Un solo proceso puede estar a la vez en la CPU, pero si un ordenador tiene por ejemplo 4 cores, ¿Sigue teniendo esta limitación? Razona tu respuesta.

No. Ya no seguiría teniendo esta limitación, ya que con 4 cores, podría realizar 4 procesos diferentes al mismo tiempo.

Los cores son, como un subprocesador en sí mismo.

Como un núcleo es un procesador en sí mismo, una CPU multinúcleo de dos núcleos pueda ejecutar dos tareas al mismo tiempo.

Una CPU con dos núcleos sí que podría realizar dos tareas al mismo tiempo, pero no más. Uno de cuatro, pues cuatro, y así de forma correlativa con tantos núcleos como incorpore.

Se puede definir como el flujo de control de programa.

Aunque un núcleo solamente pueda realizar una tarea al mismo tiempo, se pueden usar los hilos para hacer creer al usuario (y al propio ordenador) que sí se puede hacer más de una cosa al mismo tiempo.

En vez de realizar una tarea por completo, se divide la tarea en porciones (cada hilo se encarga de un aspecto concreto del programa), de modo que

se va alternando entre porciones de tareas para que parezca que ambas se ejecutan al mismo tiempo.

Hacemos un poco de un proceso y otro poco de otro proceso; cada uno de esos trozos se corresponde con el hilo. Así, no tenemos que esperar a que una tarea acabe para comenzar otra.

El número de hilos corresponde de manera directa con el número de tareas que se pueden llevar a cabo de forma pseudoparalela (es decir, de forma 'simultánea').

10. Responde

0'25

10.1. ¿Un Sistema Operativo moderno sería capaz de funcionar si un gestor de memoria? Razona tu respuesta

Como poder, podría, pero el usuario no podría hacer nada con el ordenador ya que no dispondría de una GUI (interfaz gráfica de usuario) ni ninguna consola o terminal donde poder ejecutar comandos reconocibles, ergo como ya he dicho, no se podría hacer nada. Sin embargo esto es posible, recordando el ejemplo de las antiguas consolas de cartucho, como la GameBoy, la cual era puro hardware, y el cartucho del juego también lo era, ya que éste contenía una serie de instrucciones, que al introducirlo en la GameBoy, ésta las iba ejecutando para que se ejecutase el propio juego, y ahí no había ningún SO por detrás, ya que el usuario no manejaba nada, simplemente jugaba al propio juego.

10.2. ¿Cuáles son las funciones principales del Gestor de memoria de un SO?

- Controlar las zonas de memoria que están asignadas y cuáles no.
- Asignar memoria a los procesos cuando la necesiten y retirársela cuando terminen.
- Evitar que un proceso acceda a la zona de memoria asignada a otro proceso.
- Gestionar el intercambio entre memoria principal y memoria secundaria en los casos en que la memoria principal está completamente ocupada, etc.

11. Rellena la siguiente tabla con los tipos de particionamiento de memoria que se dan, tanto en la gestión de la memoria física como en la memoria virtual.

0'20

Rellena la tabla, crea las filas y combinaciones que sean necesarias)

Memoria física			Memoria virtual Multitarea		
Monotarea	Multitarea				
	particiones Fijas	Particiones Variables	Paginación Pura	Segmentación Pura	Combinación de ambas técnicas
	absoluto/relocalizable				

12. Explica con tus palabras y dibuja un gráfico sobre cómo funciona la asignación de memoria en un SO con gestión de memoria de particiones fijas y estáticas.

0'80
✓✓

Se divide el espacio de memoria en particiones fijas, ya predefinidas antes de que lleguen los procesos. El número de particiones se mantiene fijo en el tiempo, así como el tamaño de cada una de las particiones.

Existen 2 maneras de hacer esto:

a. Una cola por partición:

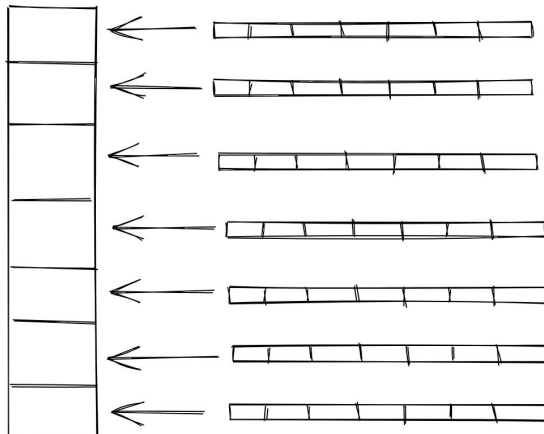
Se tiene una cola por cada partición y se coloca cada trabajo en la cola de la partición más pequeña en que quepa dicho trabajo, planificando cada cola por separado.

b. Una única cola común a todas las particiones.

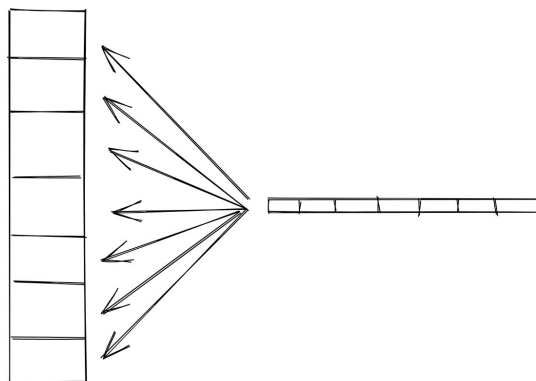
Se tiene una única cola común para todas las particiones. El sistema operativo decidirá en que partición se ubica cada proceso.

El gestor de memoria establecerá mecanismos para impedir que un proceso pueda acceder a una zona de memoria que está fuera de la memoria correspondiente a la partición en la que se encuentra

Una cola por cada partición



Una cola única para todas las particiones



13. Responde, en la gestión de la memoria con particiones fijas y estáticas...

0'33

13.1. ¿Cómo se llama al problema de que existe cierta memoria libre que no se puede utilizar para asignar a otros procesos?

Fragmentación interna

13.2. ¿Porqué se produce el fenómeno de la pregunta anterior? ¿Cómo se podría solucionar?

Este fenómeno ocurre cuando tenemos áreas de memoria ocupadas intercaladas con áreas de memoria libres.
Esto se podría solucionar con las particiones variables.

Desfragmentación

- 13.3. ¿Qué significan los conceptos particiones fijas? ¿Y a que se refiere el concepto de estáticas?

El concepto de particiones fijas se refiere a que la memoria se divide en particiones de igual tamaño antes de que lleguen los procesos, y con estáticas se refiere a que éstas no varían en su tamaño ni número con el paso del tiempo y la creación/finalización de los procesos.

14. Responde, en la gestión de la memoria con particiones variables...

- 14.1. ¿Se puede aprovechar toda la memoria libre disponible para asignar a procesos? Argumenta tu respuesta

No del todo.
Es cierto que la memoria se aprovecha mejor que con las particiones fijas, pero igualmente acaba apareciendo fragmentación.

- 14.2. ¿Cómo se denomina el fenómeno producido en este tipo de particionamiento que impide que se asigne toda la memoria libre a los procesos?

Se trata de la fragmentación externa, la cual aparece cuando en el último tramo de memoria libre disponible, no cabe el siguiente proceso que se fuera a ejecutar, con lo que éste último no puede ejecutarse.

- 14.3. ¿Existe alguna solución definitiva? ¿Y parcial?

La solución a la fragmentación externa sería la Paginación, pero esto tan sólo sería una solución parcial.

15. La gestión de memoria con particiones variables fue una mejora sustancial respecto a las particiones fijas, al igual que supuso la introducción de la paginación en la gestión de la memoria. Explica ...

- 15.1. ¿En qué consiste la gestión de la memoria paginada?

Consiste en dividir la memoria en partes iguales llamadas "marcos de página" antes de que lleguen los procesos y, a su vez, los procesos se dividen en partes más pequeñas llamadas "páginas", de modo que se van cargando las diversas páginas de los procesos en los marcos de página correspondientes.

Nombre:

- 15.2. ¿Qué conceptos clave fueron introducidos en este sistema de particionamiento?

- 15.3. ¿Es posible utilizar toda la memoria libre disponible en este sistema de particionamiento? En caso de que tu respuesta sea no, indica donde reside la memoria libre no utilizable por los procesos.

Con la paginación, volvemos al problema de la fragmentación...
Cuanto más grandes sean las páginas, más fragmentación interna aparecerá en el último marco de página de un proceso.

16. La segmentación es una evolución de la gestión de la memoria paginada, ...

- 16.1. ¿Qué diferencias más importante existen entre el concepto de página y segmento?

- La diferencia básica entre la paginación y la segmentación es que una página siempre tiene un tamaño de bloque fijo, mientras que un segmento tiene un tamaño variable.
- La paginación puede llevar a una fragmentación interna ya que la página tiene un tamaño de bloque fijo, pero puede ocurrir que el proceso no adquiera el tamaño de bloque completo, lo que generará el fragmento interno en la memoria. La segmentación puede conducir a una fragmentación externa a medida que la memoria se llena con los bloques de tamaño variable.
- En la paginación, el usuario solo proporciona un único entero como la dirección que se divide por el hardware en un número de página y Offset. Por otro lado, en la segmentación, el usuario especifica la dirección en dos cantidades, es decir, número de segmento y desplazamiento.
- El tamaño de la página es decidido o especificado por el hardware. Por otro lado, el usuario especifica el tamaño del segmento.
- En la paginación, la tabla de la página asigna la dirección lógica a la dirección física y contiene la dirección base de cada página almacenada en los marcos del espacio de la memoria física. Sin embargo, en la segmentación, la tabla de segmentos asigna la dirección lógica a la dirección física, y contiene el número de segmento y el desplazamiento (límite de segmento).

- 16.2. ¿Qué fenómeno muy recurrente en los anteriores modelos de asignación se erradica con la segmentación?

La fragmentación interna.

Nombre:

17. Con la memoria virtual, un proceso puede estar en memoria secundaria cuando no se esté utilizando, pero necesita estar en memoria física si necesita volver a ejecutarse. C

Explica con tus palabras el proceso que lleva a cabo el SO cuando necesita ejecutar una o varias páginas de un proceso, y estas se encuentran en la memoria virtual. Explica cómo el sistema operativo realiza este proceso, de la forma más detallada que puedas. ND

Quando un proceso esté cargado en la memoria de forma contigua, el Instruction Pointer sabe que la próxima instrucción a la actual es la siguiente que viene... pero cuando la memoria está paginada cuando el IP llegue al final de una página, la siguiente instrucción no será la siguiente, si no que estará en otra página diferente y saltada de la actual que acaba de terminar. Para resolver el dónde debe de apuntar el IP, se creó el TLB (Translation Lookaside Buffer) el cual va traduciendo las direcciones de la memoria física en direcciones hacia la memoria virtual (secundaria), otorgándole al IP un direccionamiento relativo compuesto por un único número entero el cual comprende el nº de la página y un Offset.

18. Responde sobre la gestión de la Entrada/Salida en los SO... OS

- 18.1. ¿Qué tipo de periféricos existen según su dirección? ✓

- De entrada: reciben la información del exterior y se la pasan al procesador (teclado/ratón)
- De salida: presenta la información procesada al usuario a través de un dispositivo de salida (como el monitor)
- De entrada y salida: combinación de los dos mecanismos anteriores

- 18.2. Un periférico, para gestionar la comunicación con el SO se compone de 2 partes principales. Indica el nombre de estos componentes y cuál es su función. ✓

- Device Driver: también conocido como controlador de dispositivo, el cual se encarga de la comunicación entre el dispositivo y la CPU
- Dispositivo mecánico, electromecánico o electromagnético el cual se conecta al ordenador

- 18.3. ¿Cuál es la principal función que realiza el SO para tratar con las particularidades que presentan los periféricos? ✓

Abstraer de las peculiaridades del hardware del dispositivo y así estandarizar un uso sencillo del mismo.

19. Respecto a las técnicas de E/S que realiza el SO para comunicarse con los periféricos ...

19.1. Enuméralas y explica brevemente cada una de ellas.

a) E/S programada

- Cuando el procesador ejecuta un programa y encuentra una instrucción relacionada con la E/S (leer, escribir o comprobar el estado de un dispositivo), ejecuta esa instrucción generando uno o varios mandatos al módulo de E/S apropiado.
- El módulo de E/S realiza la acción solicitada pero no realiza ninguna acción para avisar al procesador.
- Por tanto, después de invocar la instrucción E/S, el procesador debe tomar un papel activo para determinar cuándo se completa la instrucción.
- El procesador comprueba periódicamente el estado del módulo E/S hasta que se completa la operación.
- Aquí no se ha producido ninguna interrupción, sin embargo, el procesador ha colaborado con el módulo E/S para comunicarse en dirección al dispositivo externo y no al revés.

b) E/S por interrupciones

- El problema de la E/S programada es que el procesador tiene que esperar hasta que el módulo E/S correspondiente esté listo para la recepción o transmisión de más datos.
- Mientras tanto, el procesador está esperando y debe comprobar repetidamente el estado del módulo de E/S.

A causa de esto, el rendimiento global del sistema se degrada.

- Una alternativa es que el procesador genera un mandato de E/S y acto seguido continúe realizando algún otro trabajo útil mientras que el módulo de E/S realiza su función y se comunica con el dispositivo externo.

c) Acceso directo a memoria (DMA)

La E/S dirigida por interrupciones, aunque es más eficiente que la E/S programada, igualmente requiere de la intervención activa del procesador para transferir datos entre la memoria y un módulo de E/S, ya que cualquier transferencia de datos debe atravesar un camino a través del procesador. Por lo que, ambas formas de E/S tienen dos inconvenientes:

- La tasa de transferencia de E/S está limitada por la velocidad con la que el procesador puede comprobar el estado de un dispositivo y ofrecerle servicio.
- El procesador está involucrado en la gestión de una transferencia de E/S, por lo que se deben ejecutar varias instrucciones por cada

transferencia de E/S.

19.2. Explica qué papel juega el microprocesador en cada técnica.

20. Respecto a la gestión del almacenamiento, el SO juega un papel muy importante...

o 10

20.1. ¿En qué consiste la organización lógica y física del almacenamiento? Explica ambos conceptos y qué diferencia existe entre ellos.

20.2. Los archivos y carpetas pertenecen a la organización lógica. ¿Qué metadatos más relevantes guarda el sistema de almacenamiento sobre los ficheros?

- fecha creación
- fecha modificación
- autor/propietario
- dirección origen

D

20.3. ¿Qué son rutas absolutas y relativas?

Una ruta absoluta es aquella que se realiza entre diferentes directorios; mientras que una ruta relativa se realiza dentro del mismo nivel.