

Apellido y Nombre:..... Legajo:.....

A					B					Nota	FIRMA DEL ALUMNO (Solo en caso de revisión de examen)
					Ejercicio 1			Ejercicio 2			
1	2	3	4	5	a	b	c	a	b		

A) TEORIA: Explícitamente define como VERDADERA o FALSA cada una de las siguientes afirmaciones, FUNDAMENTANDO su respuesta en no más de 3 renglones.

A1) Los archivos con organización indexada cuentan con un único índice formado por la combinación de todos los campos que maximiza la velocidad de búsqueda dentro del archivo.

Falso ya que es necesario tener un índice por cada campo sobre el que se puedan hacer búsquedas

A2) El cifrado (encriptación) con la clave pública permite asegurar la confidencialidad de los datos

Verdadero, ya que solo puede ser descifrada con una clave privada

A3) Si se trabaja con tabla de páginas invertidas es imposible que una referencia a memoria pueda generar 2 fallos de paginas

Verdadero ya que la tabla de páginas invertida indexa por número de frame en lugar de indexar por número de página.

A4) Una falla en el mecanismo de sincronización de procesos concurrentes puede generar el fenómeno de inversión de prioridades

Verdadero ya que si un proceso de menor prioridad lockea un recurso necesario por ambos procesos el proceso de mayor prioridad no podrá continuar hasta que el otro haya finalizado

A5) En una estructura SMP (SymmetricMultiProcessor) no se comparte la memoria principal para evitar problemas de coherencia de cache en los procesadores

Falso ya que la memoria principal es compartida por todos los procesadores. El SO debe implementar los mecanismos necesarios de protección para que la ejecución se haga sin problema

B) PRACTICA - Resolver los siguientes ejercicios

Ejercicio 1 En una fábrica un nuevo dueño desea conocer el tiempo de respuesta del proceso D para citar a esa hora a un grupo de empleados de control de calidad y no antes.

Los procesos que ejecutan son los siguientes:

Proceso	Llegada	1ra ráfaga anterior	CPU	I/O	CPU	I/O	CPU
A	0	5	3	3	1	4	4
B	1	4	2	1	2	2	2
C	3	5	2	1	2	4	2
D	9	10	2	5	5		

Cada número indica unidades de tiempo.

Se pide realizar el Gantt de ejecución y calcular el turnaround time de cada proceso sabiendo que el algoritmo de planificación utilizado es SRT.

SOLUCION EJERCICIO 1:

RESPUESTA UTILIZANDO ESTIMADORES

D																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				
---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ejercicio 2: Se dispone de un sistema que trabaja de la siguiente manera:

- Abre un archivo de texto y va leyendo registro por registro.
- Cada registro leído (512 bytes) es almacenado en memoria
- Cuando termina de leer el archivo, lo cierra.
- Por último, lee nuevamente los registros de la memoria principal, en orden inverso, es decir, primero lee el último registro guardado, luego el anterior y así sucesivamente hasta llegar al primero.

El tamaño del bloque del disco es de 2KB y éste proceso cuenta con 4 frames de memoria disponibles (de 512 bytes c/u).

Se pide que determine la cantidad de fallos de página que ocurrirán si el SO utiliza el método LRU y el sistema lee la totalidad de los registros del archivo, que son 10. Considere que los primeros frames se producen Page Fault.

Considere únicamente la memoria utilizada por los datos, es decir, descarte el propio código del programa.

También se pide que calcule el tiempo que demorará el sistema en completar su objetivo, sabiendo que:

- El disco gira a 7500 rpm.
- El SO utiliza el algoritmo de planificación del brazo FIFO.
- Está formateado con interleave 0.
- Tiene 8 sectores por pista y 200 pistas
- Considere que el disco tiene una única superficie.
- El tiempo para pasar entre pistas es de 2 ms.
- El primer acceso a disco se hace a la pista 5, sector 3.
- Este disco está muy fragmentado, por lo que es muy frecuente que los bloques de los archivos estén almacenados en diferentes pistas. En este archivo en particular, curiosamente los bloques se fueron almacenando cada 5 pistas, es decir: el primer bloque en la pista 5, el segundo en la pista 10, etc. Afortunadamente, en todos los casos la información está almacenada en el mismo número de sector.
- El acceso a memoria es de 0.25 ms.
- Suponga que antes de comenzar con éste proceso, el brazo del disco estaba situado en la pista 16, sector 0.

Nota: No considere en sus cálculos los tiempos insumidos por el proceso de swapping.

Para calcular la cantidad de PF consideramos:

Son 10 registros que ocupan 512 bytes. Como el frame también es de 512 bytes, entonces significa que entra un registro por página. Por lo tanto son 10 páginas. Al ser todos los registros diferentes, ocurrirán 10 PF (si consideramos como tales a los 4 primeros accesos).

A esto hay que adicionarle 6 PF más, que ocurrirán cuando el proceso haga la lectura de los datos. Son 6 PF porque los 4 primeros accesos no generan PF, porque los frames ya están cargados con las páginas que necesitamos.

Entonces, son 16 PF

En cuanto al disco, primero tenemos que armar la lista de pedidos (PES). Como los bloques son de 2KB, significa que entran 4 registros por cada uno. Entonces, con 3 bloques será suficiente para almacenar todo el archivo.

El primer bloque está en la pista 5, sector 3, y según enunciado, los dos siguientes estarán en 10,3 y 15,3.

Como el disco gira a 7500 rpm, serán 8 ms por vuelta, o bien 1 ms por cada sector recorrido.

Pista/Sector	Tiempo Búsqueda	Tiempo Latencia	Tiempo Lectura	Tiempo Total por fila
16/0 - 5/3	$11 \cdot 2 = 22 \text{ ms}$	$5 \cdot 1 = 5 \text{ ms}$	1 ms	28 ms
5/3 - 10/3	$5 \cdot 2 = 10 \text{ ms}$	$5 \cdot 1 = 5 \text{ ms}$	1 ms	16 ms
10/3 - 15/3	$5 \cdot 2 = 10 \text{ ms}$	$5 \cdot 1 = 5 \text{ ms}$	1 ms	16 ms
Tiempo por columna	42 ms	15 ms	3 ms	60 ms

El tiempo total de I/O será de 60 ms

El Final conforma 2 partes: Teoría y Práctica. Se tiene que aprobar Si o Si ambas partes con un 60% bien respondidas y correctamente resueltas, como mínimo para tener aprobado dicho FINAL.

Apellido y Nombre:..... Profesor:.....

A				B	C		NOTA	FIRMA ALUMNO (Sólo en caso de revisión de examen)
1	2	3	4		1	2		

A) Explicitamente defina como VERDADERA o FALSA cada una de las siguientes afirmaciones, FUNDAMENTANDO su respuesta en no más de 3 renglones.

1. Una solución válida a un sistema representable por el Problema de los Filósofos es controlar el grado de multiprogramación del mismo.

Verdadero: En el problema de los filósofos, si existieran N-1 filósofos dentro del sistema uno podría tomar 2 tenedores en el instante 0 evitando deadlock y starvation.

2. Los semáforos son herramientas que se emplean únicamente para resolver el problema de la exclusión mutua y para ello, basta con implementar semáforos MUTEX.

Falso: Los semáforos también permiten controlar la cantidad de instancias accesibles de un objeto y para la sincronización de procesos a los efectos de establecer el orden de ejecución de las instrucciones

3. A pesar que el procesador sólo puede ejecutar una instrucción en cada instante de tiempo, gracias a la multiprogramación, es posible que dos procesos se completen en un tiempo menor que si se ejecutaran de forma secuencial, incluso en un equipo con un solo procesador.

Verdadero: Ya que se aprovechan los tiempos de espera por entrada/salida. Esto es válido siempre que exista el DMA en el sistema.

4. Uno de los posibles motivos por los cuales en un sistema debería existir un componente responsable de gestionar el acceso de los procesos a los recursos del sistema, es que el código máquina de los procesadores normalmente no dispone de instrucciones apropiadas para acceder a dichos recursos.

Falso: el objetivo de que haya mediación en el acceso a los recursos es garantizar la disponibilidad del recurso al momento de la petición y que el usuario tenga suficientes privilegios para ejecutar la operación solicitada sobre ese recurso

B) Conteste la siguiente pregunta en forma concisa.

Realizar un diagrama en donde se explique claramente como se resuelve una dirección lógica en un esquema de tabla de páginas invertidas

C) Resuelva en forma clara y detallada los siguientes ejercicios.

1. Se tienen dos discos rígidos formando un RAID 0 + 1 con 6 cabezas, 500 pistas y 40 sectores por pista de 4kb cada uno. Los discos giran a 7500 RPM (Revoluciones por Minuto), el brazo tarda 1 ms en pasar de pista a pista y en la cola del driver de uno de los discos se encolan los siguientes pedidos ordenados en el tiempo (direcciones lógicas en decimal): 917 - 1024 - 524 - 4204. Adicionalmente, se sabe que el filesystem que se encuentra montado sobre dicho disco utiliza i - nodos con 10 punteros directos a bloques, 2 indirectos simples, 1 indirecto doble y 1 indirecto triple, con punteros de 4 bytes. Si se considera que el tamaño de un sector es igual al de un bloque, indicar:

1.1. El tamaño máximo real de un archivo y del filesystem.

1.2. El tiempo promedio de posicionamiento de la cabeza, sabiendo que el algoritmo de planificación del disco es el N-Step-SCAN con N igual a 2. Tenga en cuenta que el brazo se encuentra posicionado en la dirección física (0, 5, 4) dirigiéndose al cilindro 24.

NOTA: Para la resolución de este ejercicio tome únicamente los 3 primeros decimales sin redondear.

Datos:

Cabezas: 6
Pistas: 500
Sectores por pista: 40
Velocidad: 7500 rpm
Tiempo pista a pista: 1 ms
Puneros directos: 10
Puneros indirección simple: 2
Puneros indirección doble: 1
Puneros indirección triple: 1
Tamaño Puntero: 4 By = 32 bit

i = 3

$$b) \text{ TamañoMáximoArchivo} = \sum_{i=0}^3 cp \times (TB/TP)^i \times TB$$

$$\text{Directos} = 10 \times (4096/4)^0 \times 4 \text{ Kb} = 40 \text{ Kb}$$

$$\text{Indirecto simple} = 2 \times (4096/4)^1 \times 4 \text{ Kb} = 8192 \text{ Kb} = 8 \text{ Mb}$$

$$\text{Indirecto dobles} = 1 \times (4096/4)^2 \times 4 \text{ Kb} = 4194304 \text{ Kb} = 4 \text{ Gb}$$

$$\text{Indirecto triple} = 1 \times (4096/4)^3 \times 4 \text{ Kb} = 4294967296 \text{ Kb} = 4 \text{ Tb}$$

$$\text{Tamaño Máximo Archivo} = 40\text{Kb} + 8 \text{ Mb} + 4 \text{ Gb} + 4 \text{ Tb} = 4 \text{ Tb}$$

c) Bloques Administrativos para el archivo real (468,75 Mb) se necesita hasta los punteros de indirección doble

$$\text{Bloques} = 2 (IS) + (1 + 1024) (ID) = 1027 \text{ bloques}$$

$$\text{Espacio bloques administrativos} = 1027 \times 4 \text{ Kb} = 4 \text{ Mb}$$

$$\text{Tamaño real del archivo} = 468,75 \text{ Mb} - 4 \text{ Mb} = 464,75 \text{ Mb}$$

$$\text{Tamaño del File System} = 2^{TP} \times TB = 2^{32} \times 4 \text{ Kb} = 16 \text{ Tb}$$

$$\text{Tamaño del Filesystem real} = 468,75 \text{ Mb} - 4 \text{ Mb} = 464,75 \text{ Mb.}$$

1.2. El tiempo promedio de posicionamiento de la cabeza,

Datos:

$$\text{Orden de pedidos} = 917 - 1024 - 524 - 4204$$

Dir. lógica	Dir. física
917	(3, 4, 38)
1024	(4, 1, 25)
524	(2, 1, 5)
4204	(17, 3, 5)

$$\text{Posición} = (0, 5, 4) \uparrow$$

$$\text{Pista a pista} = 1 \text{ ms}$$

$$\text{Rotacional} = 7500 \text{ rpm una rotación} = 60.000 / 7500 = 8 \text{ ms una rotación}$$

$$40 \text{ sectores tardan } 8 \text{ ms } 1 \text{ sector} = 8 \text{ ms} / 40 \text{ sectores} = 0,2 \text{ ms}$$

Algoritmo de planificación del disco es el N-Step-SCAN con N igual a 2, el brazo se encuentra posicionado en la dirección física (0, 5, 4) ↑

$$(0, 5, 4) \rightarrow (3, 4, 38) = 3 \times 1 \text{ ms}$$

$$(3, 4, 19) = 3 \text{ ms} + 3,8 \text{ ms} = 6,8 \text{ ms}$$

$$(3, 4, 39) \rightarrow (4, 1, 25) = 1 \times 1 \text{ ms}$$

$$(4, 1, 4) = 1 \text{ ms} + 4,2 \text{ ms} = 5,2 \text{ ms}$$

$$(4, 1, 26) \rightarrow (17, 3, 5) = 13 \times 1 \text{ ms}$$

$$(17, 3, 11) = 13 \text{ ms} + 6,8 \text{ ms} = 19,8 \text{ ms}$$

$$(17, 3, 6) \rightarrow (500, X, Y) = 483 \text{ ms}$$

$$(500, X, 21)$$

$$(500, X, 21) \rightarrow (2, 1, 5) = 498 \times 1 \text{ ms}$$

$$(2, 1, 31) = 498 \text{ ms} + 2,8 \text{ ms} = 500,8 \text{ ms}$$

$$\text{TOTAL} = 1015.6 \text{ ms}$$

2. Una importante torre de Catalinas Norte posee 12 ascensores que utilizan los empleados para poder ir a sus respectivas oficinas. La capacidad de los mismos es para 14 personas cada uno y no pueden sobrepasar ese límite debido a que suena una alarma y las puertas no se cierran. Los pseudo códigos que simulan la situación son los siguientes:

Proceso_Empleado() While (1) { Llamar_Ascensor() i = Ingresar_Ascensor() Ascensores[i] = Actualizar_Piso() Salir_Ascensor(Ascensores[i]) }	Proceso_Ascensor() While (1) { Posicionarse (Ascensores[i]) }
--	--

Tenga en cuenta que:

- La función "Posicionarse" hace que el ascensor vaya al piso seleccionado por el empleado, el cual se encuentra determinado por el vector global "Ascensores", con una posición por cada ascensor en el sistema.
- La función "Ingresar_Ascensor" devuelve el número de ascensor a utilizar por el empleado.
- Los empleados no pueden bajar o subir del ascensor hasta que este no este posicionado en el piso deseado.
- Existen N empleados que pueden estar esperando por los ascensores.

Con los datos proporcionados, se pide que incluya los semáforos que crea convenientes, detallando el tipo y los valores iniciales de los mismos, para que no se produzca deadlock, ni starvation. Cabe mencionar que la solución propuesta debe ser lo más eficiente posible.

Proceso_Empleado() While (1) { Llamar_Ascensor() Down(variable_i) i = Ingresar_Ascensor() down(capacidad[i]) Ascensores[i] = Actualizar_Piso() Up(Actualizar_piso[i]) Down(sale_ascensor[i]) Salir_Ascensor(Ascensores[i]) Up(variable_i) Up(capacidad[i]) Up(ascensores) }	Proceso_Ascensor() While (1) { Down(ascensores) Down(Actualizar_piso[i]) Posicionarse (Ascensores[i]) Up(sale_Ascensor[i]) }
--	--

Condición de Aprobación: Para aprobar este examen deberá tener como mínimo 3 (tres) preguntas teóricas (Puntos A y B) y 1 (uno) ejercicio (Punto C) correctamente resueltos.

Apellido y Nombre:..... Profesor:.....

A					B			NOTA	FIRMA ALUMNO (Sólo en caso de revisión de examen)
1	2	3	4	5	1	2	3		

A) Explicitamente defina como VERDADERA o FALSA cada una de las siguientes afirmaciones, FUNDAMENTANDO su respuesta en no más de 3 renglones.

1. En un sistema, los procesos recién creados entran en una cola de espera. Entran en la cola de Ready sólo cuando se ve que la utilización de la CPU cae por debajo del 20%. Entonces se puede deducir que planificador a largo plazo utiliza este sistema.

VERDADERO: Justamente se está describiendo el algoritmo que utiliza el sistema operativo para determinar el grado de multiprogramación del sistema, lo que es controlado por el planificador a largo plazo.

2. Un estudio sobre políticas de reemplazo en sistemas paginados de memoria expone lo siguiente: "Al observar el efecto del número de frames disponibles para los procesos, para una cadena de referencia dada, se observa que al aplicar la política de Clock es posible que al aumentar el número de frames disponibles aumente el número de fallos de página". Dicho estudio describe que se produce la anomalía de Belady.

FALSO: La anomalía de Belady se da únicamente en el algoritmo FIFO.

3. En los Sistemas de procesamiento en serie solamente se producen desperdicios de procesamiento de CPU si el JOB finalizaba antes de lo estimado.

FALSO: También se desperdiciaba tiempo si el JOB no finalizaba, ya que el tiempo real de ejecución de un job podía ser mayor a lo estimado.

4. La lista de i-nodos de un sistema de archivos tipo UNIX crece a medida que se crean nuevos archivos.

FALSO: La lista de inodos es fija y se define al momento de dar formato al disco.

5. La cuarta condición de Coffman (Espera Circular) es necesaria y suficiente para la existencia de deadlock si se tiene un sistema con recursos de una sola instancia.

VERDADERO: La 4ta condición de Coffman es espera circular y con recursos de una sola instancia basta con tener espera circular para que se produzca deadlock. En cambio con recursos de una sola instancia puedo tener un círculo en el grafo de asignación de recursos y no producirse deadlock.

B) Resuelva en forma clara y detallada los siguientes ejercicios.

1. Considere un sistema operativo que implementa la siguiente política para prevenir el deadlock en el sistema: Si un proceso tiene tomado el recurso (Semáforo) C y quiere tomar el A, será abortado, liberando todos los recursos tomados, debido que A se encuentra antes en el alfabeto que C. Si se quisiera tomar E, F, ..., Z no existiría impedimento alguno. Se inicializan los valores de los semáforos en: A = E = F = U = 0, B = C = R = 1. En la siguiente tabla se detallan los procesos a ejecutar:

P0	P1	P2	P3	P4
DOWN(C)	DOWN(C)	DOWN(R)	DOWN(A)	DOWN(E)
UP(A)	DOWN(R)	UP(A)	DOWN(A)	DOWN(F)
DOWN(F)	UP(C)	DOWN(U)	UP(F)	UP(E)
DOWN(R)	DOWN(B)	DOWN(C)	UP(U)	DOWN(R)
UP(C)	UP(E)	UP(C)	DOWN(E)	UP(F)
UP(F)	DOWN(A)	UP(R)	UP(E)	UP(R)
UP(R)	UP(B)	DOWN(A)	UP(A)	DOWN(A)
UP(E)	UP(R)	UP(U)	UP(A)	UP(A)

Sabiendo que el planificador que se utiliza es Round Robin con quantum de 2 unidades de tiempo, que el orden de llegada de los procesos es P0, P1, P2, P3 y P4, que se encuentran todos en la cola de Ready y que cada operación sobre los semáforos insume 3 unidades de tiempo, se pide que resuelva la traza de sincronización, indicando que procesos finalizan y en que orden, y que procesos no finalizan y porque. Es excluyente para la aprobación de este punto que indique con un número de secuencia el orden en que se ejecutan las instrucciones. Tenga en cuenta que ante simultaneidad de eventos tiene prioridad el proceso bloqueado.

2. Sea un sistema que utiliza paginación por demanda como técnica de la administración de la memoria. En el mismo, existe un proceso en ejecución cuya tabla de páginas es la siguiente:

Nro. Página	Bit de Presencia	Tiempo Última Referencia	Modificada	Frame
0	1	11	0	7
1	1	8	1	4
2	0	2	1	4
3	0	7	0	6
4	0	4	1	6
5	1	9	0	3
6	1	10	1	6
.....
63	0

Se sabe que el total de la memoria principal reservado para este proceso es de 4KB (Sin considerar el espacio incurrido por el sistema operativo) y que las referencias a memoria son las siguientes: 4982 - 1732 - 8741 - 8422 - 1244 - 3014 (Dirección lógica en decimal). Considerando que el algoritmo de elección de la víctima es el LRU con asignación fija y alcance local, se pide que calcule cuantos bloques de disco son accedidos, sabiendo que el tamaño de un bloque es de 512 Bytes y que en página 0 reside el código del proceso. Justifique su resolución incluyendo las traducciones de las direcciones lógicas a físicas e indicando en cada instante significativo de tiempo el estado de la memoria.

3. Se tiene una arquitectura master / slave con 3 procesadores, dedicando uno de ellos a la ejecución del sistema operativo y los restantes se ejecutan los procesos de usuario con un algoritmo Round Robin con quantum de 3 unidades de tiempo y cola única. Los siguientes son los procesos a ejecutar:

Proceso	Tiempo Llegada	CPU	IO	GPU
P1	1	2	2	5
P2	0	3	3	5
P3	0	4	4	3

Se pide que determine el turnaround time (Tiempo de ejecución) de cada proceso mediante la confección en forma clara y detallada de un diagrama de GANTT.

Resolución

P0	P1	P2	P3	P4
DOWN(C)(1)	DOWN(C)(2)	DOWN(R)(3)	DOWN(A)(4)	DOWN(E)(5)
UP(A)(6)	DOWN(R)(17)	UP(A)(7)	DOWN(A)(8)	DOWN(F)(23)
DOWN(F)(9)	UP(C)(20)	DOWN(U)(10)	UP(F)(11)	UP(E)(24)
DOWN(R)(12)	DOWN(B)(22)	DOWN(C)(14)	UP(U)(13)	DOWN(R)(26)
UP(C)(15)	UP(E)	UP(C)	DOWN(E)(16)	UP(F)(28)
UP(F)(18)	DOWN(A)	UP(R)	UP(E)(25)	UP(R)(30)
UP(R)(19)	UP(B)	DOWN(A)	UP(A)(27)	DOWN(A)(31)
UP(E)(21)	UP(R)	UP(U)	UP(A)(29)	UP(A)

FINALIZAN: P0(21) y P3(29)

ABORTAN: P2(14), P1(22) y P4(31)

Estado Final Semáforos: A = 2 E = 1 F = 1 U = 1 B = 1 C = 2 R = 1

Condición de Aprobación: Para aprobar este examen deberá tener como mínimo 3 (tres) preguntas teóricas y 2 (dos) ejercicios correctamente resueltos.

Apellido y Nombre:..... Profesor:.....

A		B			C		NOTA	FIRMA ALUMNO (Sólo en caso de revisión de examen)
1	2	1	2	3	1	2		

A) Explicitamente defina como VERDADERA o FALSA cada una de las siguientes afirmaciones, FUNDAMENTANDO su respuesta en no más de 3 renglones.

1. Las memorias ROM (ReadOnlyMemory) juegan un papel importante a la hora de implementar un sistema de tiempo real.
VERDADERO: Las memorias ROM son importantes a la hora de resguardar información en sistemas de tiempo real, ya que se necesita de un dispositivo de almacenamiento de alta velocidad para cumplir con los tiempos de respuesta que estos sistemas demandan. Por lo que una solución válida es tener memorias ROM en vez de discos rígidos.

2. En un sistema distribuido, un reloj global perfecto no es posible obtenerlo, ya que los ticks de los relojes físicos locales son siempre, aunque sean bastante precisos, heterogéneos.
VERDADERO: Los relojes de cada máquina en un sistema distribuido son heterogéneos y muy difícilmente coincidan. Para lograr una sincronización de los relojes a nivel lógico o conocer el estado global del sistema es que se desarrollaron algoritmos para ese propósito como lo es el de Chandy - Lamport.

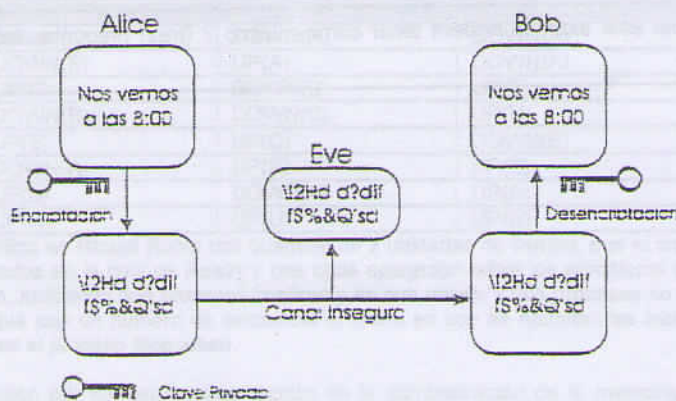
B) Conteste en forma concisa las siguientes cuestiones.

1. Bajo el seudónimo de MORPHEO, usted se va a convertir en un hacker en una cruzada personal contra los profesores de la materia. El grupo de sistemas operativos dispone de una máquina UNIX, denominada `ilws.sisop.fi.uba.ar` (ILWS son las siglas de "I love William Stallings") donde mantiene las aplicaciones y datos relativos a las calificaciones y a los exámenes que se van a tomar en futuras fechas. El Jefe de Cátedra se ha enterado de esta situación y ha propuesto la implementación de un algoritmo de encriptación de datos para transmitir el examen final a los ayudantes para su resolución. ¿Con qué tipo de algoritmo de encriptación de datos se estima que se encontraría cuando intente descifrar el contenido de la trama si únicamente se considera que la rapidez en la encriptación es relevante? ¿Por qué? Detalle un diagrama donde se describa el procedimiento de cifrado de datos en el esquema de encriptación elegido.

El esquema de cifrado que elegiría es de clave simétrica, donde la velocidad en el cifrado de los datos es una de sus ventajas en comparación con los algoritmos de claves asimétricas. Si bien una de las desventajas es el traspaso de la clave, al ser los ayudantes de Cátedra limitado a un número finito de personas y pudiendo estar físicamente en el mismo lugar al momento del traspaso, no es un inconveniente alguno el tema de cómo se informa dicha clave.

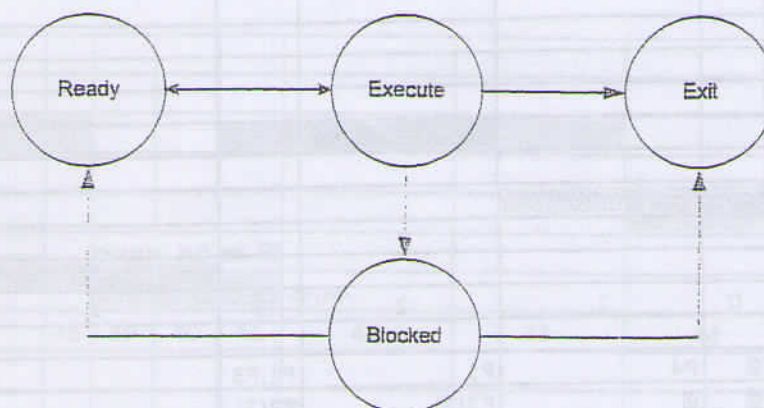
A continuación se detalla un ejemplo:

Criptografía de Clave Privada



2. Usted forma parte de un equipo de desarrollo que se encuentra diseñando un sistema operativo para smartphones, el cual se llamará "Androide" y pretende competir con otro sistema operativo de nombre similar. En lo que respecta al kernel del sistema, se definió que el mismo soportará multiprogramación de manera ilimitada, tendrá memoria virtual y dispondrá de dispositivos de E/S como por ejemplo la antena del celular. En cuanto a los procesos, se pretende que ninguno pueda monopolizar la CPU, y se debe contemplar la existencia de procesos más importantes que otros, sin que eso genere starvation (Inanición). Se espera que una vez finalizado un proceso, se puedan tomar estadísticas sobre su ejecución antes que el mismo sea eliminado completamente del sistema. Le ha tocado a usted la siguiente tarea: Proponer un diagrama de como máximo 4 estados para los procesos, explicando brevemente la causa de cada transición entre ellos e indicar que algoritmo de planificación de corto plazo serviría para este sistema (si es necesario efectuarle alguna modificación, méncionala brevemente.)

Según los requerimientos funcionales del enunciado debería optar por un planificador a corto plazo por prioridades **PREEMPTIVE**. Como se necesita que no se produzca Starvation en el sistema, una solución válida es implementar **AGING**, donde los procesos que tengan un tiempo de no utilización de la CPU se le incrementa su prioridad para que puedan ejecutar. El siguiente es un diagrama válido para responder a los requerimientos:



Estados:

- Ready: Se encolan los procesos listos para ser ejecutados con algoritmo por prioridades y AGING.
- Execute: Proceso que se encuentra efectivamente haciendo uso de la CPU.
- Blocked: Procesos en espera o haciendo uso de la Entrada / Salida. Una por cada dispositivo.
- Exit: Estado en el cual se puede tomar estadísticas antes de que el proceso sea eliminado completamente del sistema.

3. En un sistema con multiprocesamiento, para planificar procesos se puede utilizar una única cola de ready compartida por todos los procesadores, o bien que cada procesador disponga de su propia cola de Ready con asignación estática. ¿Qué ventajas e inconvenientes tiene cada una de estas dos técnicas? En un equipo tipo PC, ¿qué sistema implementaría? Justifique su respuesta.

Si cada procesador dispone de su propia cola de Ready, debemos gestionar el problema de que pueden darse desequilibrios en la carga de trabajo: un procesador podría llegar a quedarse ocioso, sin procesos en cola, mientras que otro podría seguir cargado. Este problema no ocurriría con una cola única, ya que cada vez que un procesador quedara libre, tomaría uno de los procesos de la cola única y por tanto no se daría la situación de tener a un procesador más cargado de trabajo que otro. Por otra parte, el esquema de la cola única tiene el inconveniente de que se trata de un recurso compartido por todos los procesadores y hay que controlar el acceso simultáneo a la estructura, para evitar que se corrompa o que ocurran usos indebidos (p.ej. que varios procesadores decidan ejecutar el mismo proceso). Este inconveniente es tanto más grave cuanto mayor sea el número de procesadores del sistema y más frecuentes sean los cambios de contexto. En un PC multiprocesador y con la tecnología actual tenemos pocos procesadores (dos, cuatro, a lo sumo ocho), así que probablemente el uso de una cola única es más eficiente, ya que no hay demasiada competencia por la cola.

C) Resuelva en forma clara y detallada los siguientes ejercicios.

1. Dentro del plan de optimización de los servidores que dan servicio al Home Banking de un importante banco multinacional, se ha detectado que el actual algoritmo de planificación de procesos que se encuentra implementado en dichos servidores penaliza a los procesos de usuario (aquellos que hacen uso de burst de CPU y frecuentes operaciones cortas de E/S) cuando se están procesando las solicitudes de pago de sueldos de los clientes que están adheridos al servicio de pago de sueldos (con mucho tiempo de utilización de la CPU y poca entrada / salida). El resultado es que cuando un cliente efectúa una operación por Home Banking tiene la sensación que se vuelve extremadamente lento. Para resolver este problema, se han propuesto dos algoritmos de planificación (algoritmo 1 y algoritmo 2, detallados a continuación) y una batería de pruebas, compuesta por los procesos P1, P2, P3 y P4 que se describen más adelante, con el fin de probar el funcionamiento de los algoritmos. Los algoritmos son los siguientes:

Algoritmo 1:

- Se tienen dos colas, una gestionada con Round Robin con quantum igual a 2 y otra gestionada con FCFS.
- La cola gestionada con Round Robin es más prioritaria que la FCFS. La política de planificación entre colas es PREEMPTIVE con prioridades.
- Todos los procesos comienzan su ejecución en la cola 1.
- El paso a la cola FCFS se efectuará cuando los procesos finalizan el primer Quantum.
- Cuando un proceso de la cola menos prioritaria es expulsado por la llegada de un proceso más prioritario, se colocará al final de su cola.
- Finalizada la E/S de un proceso, pasa a la cola de Round Robin.

Algoritmo 2:

- Se tiene una única cola gestionada mediante SRT.

El set de procesos a ejecutar es el siguiente:

Proceso	T. Llegada	Tipo	CPU	IO	CPU
P1	2	Usuario	2	6	1
P2	0	Sueldos	4	2	3
P3	3	Usuario	1	1	2
P4	1	Sueldos	3	3	5

NOTA: Si coinciden varios eventos en un mismo instante de tiempo suponga que estos suceden en el siguiente orden:

1. Fin del quantum / Salida del proceso de la CPU.
2. Un proceso termina una operación de E/S sobre el dispositivo.
3. Llegada de un nuevo proceso al sistema.

Si se sabe que cada cambio de proceso consume 2 unidades de tiempo y suponiendo que todas las operaciones de E/S se realizan sobre el mismo dispositivo (y este es de uso exclusivo), se pide que determine el algoritmo más óptimo a implementar en función del rendimiento del sistema. Justifique su respuesta mediante la confección en forma clara y detallada de un diagrama de GANT.

[illegible]

Frame	0*	1	2	3	4	5	6	7
Proceso	1	1	1	-	2	2	-	-
Página	0	2	4	-	0	1	-	-
Bit de Referencia	1	1	1	-	1	1	-	-

Hilo	Referencia
H12	001640
H21	001880
H11	005400
H21	007770
H11	0044CC

Suponiendo que el sistema operativo ha asignado los frames 0 - 3 a los hilos del proceso P1 y los frames 4 - 7 a los hilos del proceso P2 manteniendo una política de alcance global, se pide:

Referencias a páginas:

Clock:

En total 8 page faults, 5 producen la ejecución del algoritmo de sustitución.

Referencias a páginas:

Clock:

	H22	H12	H22	H21	H21	H11	H12	H21	H11	H21	H11
	2	1	3	2	4	7	1	1	5	7	4
F0	*P1 0,1	*P1 0,1	*P1 0,1	*P1 0,1	*P1 0,1	P1 7,1	P1 7,1	P1 7,1	P1 7,1	P1 7,1	P1 7,1
f1	P1 2,1	P1 2,1	P1 2,1	P1 2,1	P1 2,1	*P1 2,0	*P1 2,0	*P1 2,0	P1 5,1	P1 5,1	P1 5,1
f2	P1 4,1	P1 4,1	P1 4,1	P1 4,1	P1 4,1	P1 4,0	P1 4,0	P1 4,0	*P1 4,0	*P1 4,0	*P1 4,1
f3		P1 1,1	P1 1,1	P1 1,1	P1 1,1	P1 1,0	P1 1,1	P1 1,1	P1 1,1	P1 1,1	P1 1,1
f4	P2 0,1	P2 0,1	P2 0,1	*P2 0,1	P2 4,1	P2 4,1	P2 4,1	P2 4,1	P2 4,1	P2 4,1	P2 4,1
f5	P2 1,1	P2 1,1	P2 1,1	P2 1,1	*P2 1,0	*P2 1,0	*P2 1,0	*P2 1,1	*P2 1,1	P2 1,0	P2 1,0
f6	P2 2,1	P2 2,1	P2 2,1	P2 2,1	P2 2,0	P2 2,0	P2 2,0	P2 2,0	P2 2,0	P2 7,1	P2 7,1
f7			P2 3,1	P2 3,1	P2 3,0	P2 3,0	P2 3,0	P2 3,0	P2 3,0	*P2 3,0	*P2 3,0
	PF	PF	PF		PF	PF			PF	PF	

En total 7 page faults. 4 producen la ejecución del algoritmo de sustitución.

Dado que el rendimiento de un algoritmo de sustitución es determinado por la cantidad de fallos de página que genera sobre una muestra de referencias a memoria, el algoritmo Clock asignación fija y alcance local presenta un mejor rendimiento que el Clock alcance global.

Condición de Aprobación: Para aprobar este examen deberá tener como mínimo 3 (tres) preguntas teóricas (Puntos A y B) y 1 (uno) ejercicio (Punto C) correctamente resueltos.

Pregunta #7:

El Dr. Brown no está completamente convencido de la eficacia de su solución, para cual calculará el tiempo que se tarda en servir las solicitudes de lectura en el RAID si se sabe que cada cilindro recorrido por el brazo tarda 0.1 ms.

Se espera que pueda ayudar a nuestros amigos en su difícil tarea, para lo cual dispondrá de tan sólo 90 minutos. To be continued in may... Thanks triple P.

El número de sectores que ocupa un bloque es igual a 4, ya que tengo sectores de 512 bytes y los bloques son de 2048 bytes.

7200 Vueltas ————— 60 segundos.

1 Vuelta ————— $x = 0,008$ segundos = 8 ms.

128 Sectores ————— 8 ms

1 Sector ————— $x = 0.0625$ ms.

Al llegar alternativamente las solicitudes a cada disco, D0 tiene las siguientes: 1, 10, 100, 200, 5 y D1 las siguientes: 20, 4, 50, 3, 0.

Las direcciones físicas de cada dirección lógica son:

D0		D1	
Registro	Dirección Física	Registro	Dirección Física
130	(1,0,3)	2566	(20,0,7)
1294	(10,0,16)	535	(4,0,25)
12844	(100,1,13)	6423	(50,0,24)
25666	(200,2,3)	393	(3,0,10)
717	(5,2,14)	100	(0,3,5)

Aplicando el algoritmo de planificación C-LOOK a cada uno, sabiendo que D0 tiene su cabeza de lectura en la dirección (20,0,1) y D1 en el (40,0,1) y que los brazos están ascendiendo tenemos para cada disco la siguiente secuencia:

D0: 20-100-200-1-5-10 — $180+199+9 = 388$ cilindros recorridos

D1: 40-50-0-3-4-20 — $10+50+20=80$ cilindros recorridos

Al ser un RAID 1 se tardará en servir todas las solicitudes el mayor de los dos tiempos, que en este caso es obviamente el D0. El tiempo será la suma de los desplazamientos de la cabeza a través de los cilindros más el tiempo de latencia y el de transferencia en cada lectura, resultando:

(20,0,1) \rightarrow (100,1,13) = $80 \times 0,1$ ms = 8 ms.
(100, 1,1) = 8 ms + (12 sectores) + 1 Lectura.

(100,1,14) \rightarrow (200,2,3) = $100 \times 0,1$ ms = 10 ms.
(200,2,46) = 10 ms + (64 Sectores) + 1 Lectura.

(200,2,4) \rightarrow (1,0,3) = $199 \times 0,1$ ms = 19,9 ms.
(1,0,67) = 19,9 ms + (64 Sectores) + 1 Lectura.

(1,0,4) \rightarrow (5,2,14) = $4 \times 0,1$ ms = 0,4 ms.
(5,2,11) = 0,4 ms + (3 Sectores) + 1 Lectura.

(5,2,15) \rightarrow (10,0,16) = $5 \times 0,1$ ms = 0,5 ms.
(10, 0,23) = 0,5 ms + (121 Sectores) + 1 lectura.

Total = 8 ms + 10 ms + 19,9 ms + 0,4 ms + 0,5 ms + (285 Sectores \times 0,0625 ms) + 5 lecturas \times 0,0625 ms = 56,925 ms.

Condición de Aprobación: Para aprobar este examen deberá tener como mínimo 3 (tres) preguntas teóricas (Preguntas 1 a 5) y 1 (uno) ejercicio correctamente resuelto (Preguntas 6 y 7).

ACL: Matriz divididas por objetos

Flybird.gif = {(sam,rw),(mktg,rw),(others,r)}

Matrix.mov, pibeschorros.mov, mundial.mov = {(Pablito,rwx),(mktg,rwx),(others,rw)}

.profile = {(udbxp02,rw),(udbxp,r)}

.sh_history = {(udbxp02,rw)}

.profile = {(udbxp02,rwx),(udbxp,r)}

CL: Matriz dividida por sujetos

Sam = {(flybird.gif,rw)}

Pablito = {(matrix.mov,rwx),(pibeschorros.mov,rwx),(mundial.mov,rwx)}

Udbxp02 = {(profile,rw),(sh_history,rw),(ssh2,rwx)}

Mktg = {(flybird.gif,rw),(matrix.mov,rwx),(pibeschorros.mov,rwx),(mundial.mov,rwx)}

Udbxp = {(profile,r),(ssh2,rw)}

Others = {(flybird.gif,r),(matrix.mov,rw),(pibeschorros.mov,rw),(mundial.mov,rw)}

Pregunta #8: Sabiendo que el proceso de login tiene asignados 3 frames de memoria inicialmente vacios, que el código del proceso ocupa solamente la pagina 0 y que el sistema utiliza segmentación paginada con direcciones de 4 bytes de los cuales 16 bits son para referenciar el segmento y 8 bits para páginas, indicar la cantidad de page faults que ocurrieron, señalando los mismos al igual que el estado final de la memoria.

0x10001 - Seg: 0x0001=1 Pag: 0x00=0 Offset: 0x01=1

0x10418 - Seg: 0x0001=1 Pag: 0x04=4 Offset: 0x18=24

0x1051E - Seg: 0x0001=1 Pag: 0x05=5 Offset: 0x1E=30

0x104C0 - Seg: 0x0001=1 Pag: 0x04=4 Offset: 0xC0=192

0x101F0 - Seg: 0x0001=1 Pag: 0x01=1 Offset: 0xF0 = 240

0x10417 - Seg: 0x0001=1 Pag: 0x04=4 Offset 0x17= 23

	0	4	5	4	1	4
Frame 0	0	0	0	0	0	0
Frame 1		4	4	4	4	4
Frame 2			5	5	1	1
	X	X	X		X	

Se producen 4 page faults. El primer page fault genera la carga del código y ocupa el frame 0 por lo que ese frame no puede utilizarse por lo que las victimas siempre se elegirán entre el frame 1 y el frame 2.

Condición de Aprobación: Para aprobar este examen deberá tener como mínimo 3 (tres) preguntas teóricas (Preguntas 1 a 5) y 2 (dos) ejercicios correctamente resueltos (Preguntas 6 a 8).

Apellido y Nombre:.....

A			B		C			NOTA	FIRMA ALUMNO (Sólo en caso de revisión de examen)
1	2	3	4	5	Ej. 1	2	3		

A) Explicitamente defina como VERDADERA o FALSA cada una de las siguientes afirmaciones, FUNDAMENTANDO su respuesta en no más de 3 renglones.

1) La inversión de prioridades es el fenómeno por el cual el sistema operativo decide invertir el orden de los procesos encolados en la Reddyqueue.

FALSO

2) Una de las ventajas de la memoria virtual es que deja de ser necesaria la traducción de direcciones lógicas a físicas.

FALSO

3) Los abrazos mortales (Deadlocks) son consecuencia de la existencia de herramientas de sincronización, con lo cual es válido afirmar que sin sincronización no existe Deadlock.

VERDADERO

B) Desarrollar brevemente:

4) Explique por qué las primitivas Down() y Up() deben ser atómicas.

5) Explique brevemente la noción de DMA, sus ventajas y si tiene alguna relación con las interrupciones.

C) Conteste en forma concisa las siguientes cuestiones.

Ejercicio 1: se debe considerar un sistema de memoria virtual con segmentación paginada, el número máximo de segmentos es 8. El tamaño máximo de segmento es 8 Kbytes y el tamaño de página es de 1 Kbytes. La memoria real disponible es de 4 frames de página.

Hay un proceso que consta de dos segmentos, el segmento 0 es de código y tiene un tamaño de 4 Kbytes y el segmento 1 es de datos y ocupa 4 Kbytes.

En un instante dado, el proceso realiza la siguiente secuencia de referencias a bytes (en hexadecimal):

\$0040, \$2020, \$0800, \$0C80, \$0407, \$2444, \$1000, \$0407, \$2845, \$1000, \$0407,

1) Indicar el formato de la dirección virtual que se utiliza.

2) Para cada una de las siguientes referencias que realiza el proceso, indicar a qué página y segmento pertenece

3) Cual será el comportamiento de los algoritmos de sustitución FIFO y LRU para la secuencia de referencias mencionada en el enunciado, considerando las faltas de página (PF) que se producen en cada uno de los casos y considere que las primeras páginas producen PF. Explicar los resultados.

SOLUCIONES:

1) Indicar el formato de la dirección virtual que se utiliza.

El formato de la dirección virtual es el siguiente:

Segmento → bits 15 al 13; Página → bits 12 al 10; Desplazamiento → bits 9 al 0

Respuesta: El tamaño de página es 1Kbytes, entonces se necesitan 10 bits para indicar el desplazamiento (bits de 0 a 9). Al ser el tamaño máximo de segmento 8 Kbytes, cada segmento podrá contener hasta un máximo de 8 páginas, por lo que se necesitan 3 bits para acceder a cada una de las páginas de un segmento (bits 10, 11 y 12). Como el número máximo de segmentos es 8 se necesitarán 3 bits para indicar el número de segmento.

2) Para cada una de las referencias que realiza el proceso, indicar a qué página y segmento pertenece.

Para cada dirección se determina el segmento y la página a la que pertenece:

Referencia	#seg	#page	offset	Segm.	page
\$0040;	0	0	0	0	0
\$2020;	0	0	1	0	0
\$0800;	0	0	0	0	2
\$0C80;	0	0	1	0	3
\$0407;	0	0	1	0	1
\$2444;	0	0	1	0	1
\$1000;	0	0	0	0	4
\$0407;	0	0	1	0	1
\$2845;	0	0	1	0	2
\$1000;	0	0	0	0	4
\$0407;	0	0	1	0	1

3) Cual será el comportamiento de los algoritmos de sustitución FIFO y LRU para la secuencia de referencias mencionada en el enunciado, considerando las faltas de página (PF) que se producen en cada uno de los casos. Explicar los resultados.

Comportamiento del algoritmo FIFO:

Referencia	0.0	1.0	0.2	0.3	0.1	1.1	0.4	0.1	1.2	0.4	0.1
Frame 1	0.0	1.0	0.2	0.3	0.1	1.1	0.4	0.1	1.2	0.4	0.1
Frame 2		0.0	1.0	0.2	0.3	0.1	1.1	1.1	0.4	0.4	0.4
Frame 3			0.0	1.0	0.2	0.3	0.1	0.1	1.1	1.1	1.1
Frame 4				0.0	1.0	0.2	0.3	0.3	0.1	0.1	0.1
Page Fault	PF	PF	PF	PF	PF	PF	PF		PF		

Comportamiento del algoritmo LRU:

Referencia	0.0	1.0	0.2	0.3	0.1	1.1	0.4	0.1	1.2	0.4	0.1
Frame 1	0.0	1.0	0.2	0.3	0.1	1.1	0.4	0.1	1.2	0.4	0.1
Frame 2		0.0	1.0	0.2	0.3	0.1	1.1	0.4	0.1	1.2	0.4
Frame 3			0.0	1.0	0.2	0.3	0.1	1.1	0.4	0.1	1.2
Frame 4				0.0	1.0	0.1	0.3	0.3	1.1	1.1	1.1
Page Fault	PF	PF	PF	PF	PF	PF	PF		PF		

Generalmente se producen más faltas de páginas con FIFO que con LRU, esto es porque el algoritmo FIFO no tiene en cuenta el comportamiento de los programas.

Se ve que se generan muchas faltas de páginas cuando comienza a ejecutarse el proceso independientemente del algoritmo que se trate. Esto es normal porque cuando empieza a ejecutarse un proceso no se encuentra cargado su conjunto de trabajo. Una vez cargado el conjunto de trabajo del proceso, el número de faltas de página disminuye con el algoritmo LRU a diferencia de FIFO, esto es porque LRU está basado en el conjunto de trabajo de los procesos y FIFO no lo tiene en cuenta.

Ejercicio 2: Sea un Sistema Operativo que utiliza un planificador con algoritmo Round Robin de quantum 4 (cuatro) y que dispone de las funciones sincrónicas Leer, Escribir y ejecutar así como también las funciones atómicas Up y Down. Conociendo que las colas de los semáforos y la única cola de Entrada/Salida son independientes y como están inicializados los semáforos, indique la traza de ejecución y el orden de terminación de los siguientes tres procesos:

Semáforos
(valor inicial)
A = 0
B = 0
C = 1

Funciones		
(tiempo en quatoms)	CPU (al inicio)	I/O (al final)
UP / DOWN	2	0
LEER / ESCRIBIR	1	3
EJECUTAR	3	0

Proceso 1	Proceso 2	Proceso 3
DOWN <A>	DOWN 	DOWN <C>
UP <C>	EJECUTAR <func2>	LEER <ArchivoA, x>
LEER <ArchivoA, y>	UP <A>	UP
DOWN <A>	ESCRIBIR <ArchivoA, y>	EJECUTAR <func2>
EJECUTAR <func2>	EJECUTAR <func1>	ESCRIBIR <ArchivoA, x>
UP 	DOWN 	EJECUTAR <func1>
DOWN <C>	UP <C>	UP <A>

P1	C	C	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa	Sa
P2			C	C	Sb	Sb	Sb	Sb	Sb	Sb	Sb	Sb			C	C	C
P3					C	C	C	B	B	B	C	C	C	C			

P1	Sa			C	C	C	B	B	B					C	C	C	C
P2	C						C	EB	EB	B	B	B					C
P3		C	C	B	B	B		C	C	C	C	C	T = 1				

P1				C	C	C	C	C	T = 2								
P2	C	C	C	Sb	Sb	Sb			C	C	T = 3						
P3																	

NOTA: La "C" es porque sigue ejecutandose porque se trata de una función atómica (UP/DOWN)

Ejercicio 3: Dado un servidor que posee cinco discos rígidos de 23GB cada uno, formando con ellos un RAID tipo 5, en el que se posee el File System HP(High Performance). La única restricción de este file system basado en inodos (con dos punteros directos, uno indirecto y otro doble indirecto) es que puede direccionar como máximo 10.000.000 bloques de datos de 12 KB cada uno.

- a) ¿Cuál es el tamaño máximo de una partición de este file system en el servidor?
b) ¿Cuál es el tamaño máximo teórico de un archivo?

a)

Teorico FS: $10.000.000 \text{ bloques} \times 12 \text{ KB} = 114,44 \text{ GB}$

Disponible FS: $4 \times 23 \text{ GB} = 92 \text{ GB}$

Respuesta Correcta = 92 GB

b)

Calculos auxiliares:

Tamaño Puntero: 24bits (con 23bits no alcanzan) = 3B

Tamaño Bloque: 12KB

Un Inodo puede referenciar a:

DIRECTOS = 2 bloques

INDIRECTO = $(12 \text{ KB} / 3 \text{ B}) = 4.096 \text{ bloques}$

Total = 4.098 bloques

Tamaño Max Teorico Archivo = $4.098 \text{ bloques} \times 12 \text{ KB} = 48 \text{ MB}$

c) Tam Max Dir = $4.098 \text{ inodos} \times 48 \text{ MB} = 196.704 \text{ MB} = 192 \text{ GB}$

Condición de Aprobación: Para aprobar este examen deberá tener como mínimo 3 (tres) preguntas teóricas (Puntos A y B) y 1 (uno) ejercicio (Punto C) correctamente resueltos.