



Nota:

Apellido y Nombre				Profesor		Tomé conocimiento de la nota: (Sólo aplazos)	

Preguntas teóricas					Ejercicios	
1	2	3	4	5	1	2

- A) **Teoría:** Explícitamente defina como **VERDADERA** o **FALSA** cada una de estas afirmaciones justificando brevemente.
- 1) En un sistema en el cual los procesos deben solicitar todos los recursos que van a utilizar al iniciarse, nunca ocurrirá un interbloqueo entre los mismos.
- 2) Todos los algoritmos de planificación tipo Feedback (colas multinivel retroalimentadas) que implementan prioridades entre colas pueden generar Inanición.
- 3) En una aplicación tipo servidor que utiliza una biblioteca externa con posibles memory leaks lo ideal es crear un hilo por cada solicitud para mitigar el impacto de dicha situación.
- 4) En un sistema que utiliza EXT, es posible que un acceso directo a un byte requiera de 4 accesos a bloques diferentes.
- 5) Utilizar paginación multinivel en un sistema con memoria virtual permite que el tamaño ocupado por tablas de páginas en memoria RAM sea en general menor comparado con utilizar paginación tradicional.

B) **Práctica:** Resuelva los ejercicios justificando las respuestas

1) En un Sistema Operativo que planifica utilizando el algoritmo SJF (sin desalojo) corren dos procesos P1 y P2 que necesitan acceder a un conjunto de archivos en común mediante lock de escritura (exclusivo) y no los cierran durante su ejecución. Estos archivos no están inicialmente abiertos por ningún proceso y cada acceso a ellos consume 1ut más 1ut extra si nunca fue abierto aún. Se sabe que el dispositivo de E/S no permite accesos en paralelo y ambos procesos se encuentran en la cola de listos en el instante 0.

		CPU	E/S	CPU	E/S	CPU
P1	KLT 1	1	arch1	1	arch2	1
	KLT 2	3	arch2	2	arch1	1
P2	KLT 3	2	arch2	2	arch1	1

- a) Realice el diagrama de gantt e Indique los tiempos de finalización de cada hilo y/o todo problema que ocurra.
- b) ¿En qué instante cambiaría el diagrama si los archivos fueran abiertos mediante lock de lectura (compartido)?

2) Un sistema maneja memoria virtual utilizando paginación bajo demanda, con direcciones lógicas de 16 bits, asignando hasta 4 frames por proceso, con sustitución local y algoritmo de reemplazo LRU. La tabla de páginas del proceso A es la siguiente:

Página	0	1	2	3	4	5	6	7	8	9
Frame	5	-	-	9	7	1	-	-	-	-
Últ. ref	200	-	-	400	300	100	-	-	-	-

- a) Indique la traza de ejecución, señalando los fallos de página que ocurren si el proceso A realiza las siguientes referencias: A000h, 0300h, 3500h, 1000h, 4100h y A001h
Se sabe que todas las referencias son válidas y la primera (A000h) no produce PF. Además, los frames se cargaron en orden ascendente.
- b) En qué referencia cambiaría la traza de ejecución si se utilizara FIFO como algoritmo de reemplazo? Justifique sin volver a realizar el ejercicio.



Nota:

Resolución Teoría

- 1) Verdadero. Se estaría implementando una estrategia de Prevención, al solicitar la totalidad de los recursos al iniciar cada proceso, impide que se cumpla una de las cuatro condiciones necesarias (retención y espera), por lo que no podría ocurrir interbloqueo.
- 2) Falso. El algoritmo Virtual Round Robin es un ejemplo de algoritmo tipo Feedback que no presenta Inanición ya que los procesos nuevos entran por la cola de menor prioridad. Otro ejemplo que no genere Inanición podría ser que implemente prioridades con Aging.
- 3) Falso. Crear un hilo por cada solicitud no minimizaría el impacto. Lo ideal sería crear un proceso por solicitud o grupo de solicitudes para que al finalizar se liberen los recursos y el memory leak afecte lo menos posible.
- 4) Verdadero. Si el bloque se encuentra en una indirección triple requerirá 4 accesos a bloques.
- 5) Verdadero. Paginación multinivel permite paginar las tablas de páginas y solo cargar las que están siendo utilizadas en cierto momento, por lo cual el tamaño ocupado en memoria RAM por las mismas suele ser menor que en paginación.

Práctica

1) a)

P1	K1		A1	A1							P1 y P2 quedan en deadlock esperando por A2 y A1 respectivamente							
	K2																	
P2	K3				A2	A2												
	0	1	2	3	4	5	6	7	8	9	14	15	16	17	18	19	20	21

b) El diagrama cambiaría en el instante 5 ya que P1 podría abrir el archivo 2 y realizar su E/S (de 1ut). Podría justificarse también que en realidad la ejecución cambia en el instante 4 ya que P1 abre A2 en ese momento y no comienza su operación de E/S solamente debido a que el dispositivo no permite accesos en paralelo.

2) a) Lo primero que necesitamos saber es qué porción de la dirección corresponde a la página y cuál al desplazamiento. Sabiendo que A000h no produce PF:

A000h -> 1010 0000 0000 0000 -> La única manera de dividir la dirección y no tener PF es tener 3 bits para página y 13 para el desplazamiento, de esta manera nos quedarían los primeros 3 bits (101) como página 5, entonces las referencias son:

A000h -> página 5, 0300h -> página 0, 3500h -> página 1, 1000h -> página 0, 4100h -> página 2 y A001h -> página 5

	5	0	1	0	2	5
Fr 1	5	5	5	5	5	5
Fr 5	0	0	0	0	0	0
Fr 7	4	4	1	1	1	1
Fr 9	3	3	3	3	2	2
			PF		PF	

b) El algoritmo FIFO, a diferencia de LRU, reemplaza las páginas según su tiempo de carga sin importar cuántas veces se haya accedido posteriormente, por lo que la referencia a la página 1 (3500h) causaría un PF pero reemplazaría a la página 5 (Frame 1).