



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 con paginación, la sobrepaginación es un problema que se puede dar con o sin memoria virtual tanto con sustitución de páginas local o global.
- 2) El journaling permite mantener integridad dentro de un file system.
- 3) Siempre que el algoritmo de planificación sea con desalojo, en caso de ocurrir una interrupción la atenderá y evaluará si conviene replanificar; en caso contrario (sin desalojo) se atenderán las interrupciones al bloquearse o finalizar.
- 4) El tamaño real máximo de un archivo siempre está dado por una relación entre el tamaño del bloque de datos y la cantidad de direcciones de referencia posibles.
- 5) En caso de querer hacer nuestra propia implementación de wait y signal (sin usar las syscalls que provee el SO) deberíamos usar otra estrategia extra de sincronización para garantizar la atomicidad.

B) Práctica: Resuelva los ejercicios justificando las respuestas

1) Un sistema utiliza paginación bajo demanda con asignación fija de 2 frames de 1MiB por proceso. La paginación es jerárquica de dos niveles con direcciones lógicas de 32bits (se usa igual cantidad de bits para cada nivel) y el algoritmo de sustitución de páginas es LRU. El sistema posee también una TLB cuyas entradas poseen sólo nro de página y nro de frame. Actualmente un proceso P1 está en ejecución y luego se carga un nuevo proceso P2 (aún no tiene páginas en memoria y se le asignan los frames 100 y 120). Este nuevo proceso realizará modificaciones sobre un archivo de 300MiB; para hacerlo de forma más eficiente lo mapeará a su imagen (a partir de la dirección lógica 20000000h de la misma). El FS utilizado es EXT2 con inodos que poseen 10 ptrs directos, 1 ind simple y 2 dobles (bloques 1 KiB y ptrs de 32 bits). Si se realizan las siguientes referencias a memoria: P1 123A0B34h - P1 00111001h - P2 010AA221h - P2 21000001h - P2 01022222h - P1 001ABCDEh - P2 30000001h

- Indicar para cada acceso:
- a) Si se accedió a la TLB y/o a la tabla de páginas. Indicar si se leyeron dichas estructuras o si también fue necesario actualizarlas.
- b) En caso de realizarse una operación de lectura/escritura de páginas correspondientes al archivo indicar la cantidad de accesos a bloques del FS.

Estado TLB: pág 123h - frame 20 || pág 001h frame 30

Nota: dentro del límite de 2 frames por proceso no se cuentan las páginas de la tabla de páginas (paginación jerárquica)

2) Se dispone de un sistema con planificación a corto plazo bajo el algoritmo VRR Q=3. En un determinado momento, ejecutan 4 procesos:

P1	P2	P3	P4	Datos
1	2	3	0	<u>Valores iniciales:</u> sem_t A=B=C=1; y int a=b=1 La segunda fila corresponde al tiempo de llegada de cada proceso Las operaciones wait y signal son <b>atómicas</b> mediante la deshabilitación de interrupciones y consumen 2 ut. Las operaciones <b>no atómicas</b> consumen 1 ut. Las lecturas insumen una unidad de tiempo por unidad de lectura (no ejecutan en la CPU). La función <i>sleep(X)</i> bloquea el proceso X unidades de tiempo. Funciones: read(puntero_archivo, buffer, unidades_lectura); sleep(unidades_tiempo); print(variable)
wait(A); read(file, &c, 1); a--; b--; *c=0; a=b; wait(B); print(a); print(b);	a++; b++; wait(B); b=a; read(file, &c, 2); wait(A); a++; print(b);	wait(B); wait(C); read(file, &c, 2); signal(B);	sleep(18); signal(A); signal(B); signal(B);	

Realice un diagrama de gantt y detalle el estado de cada uno de los procesos. En todos los casos que se ejecute 'print' imprima el valor.