



Nota:

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

Preguntas teóricas					Ejercicios	
1	2	3	4	5	1	2

- A) **Teoría:** Defina explícitamente como VERDADERA o FALSA cada una de estas afirmaciones justificando brevemente.
- 1) Luego de ejecutar correctamente una syscall del estilo *write(file, message, size)*, el mensaje podría no haber sido escrito a disco todavía, y en ese caso el proceso debería evitar modificar la variable “message” por existir la operación de I/O en curso.
- 2) Ante una solicitud a una porción de memoria inválida, no se requieren accesos a memoria para determinar dicha invalidez.
- 3) Previo a cada ejecución de una instrucción privilegiada, el sistema operativo interviene para verificar que el proceso tenga los privilegios adecuados.
- 4) Cuanto más pequeña sea la pila de un proceso, mayor será el grado de recursividad que podrá tener su código.
- 5) Si se compara entre un deadlock entre los procesos “P1” y “P2” y los recursos “R1” y “R2” contra un livelock (involucrando los mismos procesos y recursos), se podría concluir que el primer caso impacta menos en el sistema.

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

- 1) Un sistema utiliza paginación jerárquica dedicando igual cantidad de bits para sus dos niveles de nro de págs. Se sabe que en el sistema hay 3 marcos libres y para administrarlos utiliza un bitmap de marcos libres (1: ocupado, 0: libre), asignando el primero que esté libre. En el sistema el planificador acaba de desalojar a PB y coloca a ejecutar PA.
1. Indique la DF y pasos realizados para traducir las siguientes DLs del PA: A101111h - 030110A1h -03030000h
2. Indique
- a. La máxima fragmentación interna que podría tener un proceso en dicho sistema
- b. ¿Cuál es el estado final de la TLB? ¿Qué hubiese cambiado si le sacamos la última columna a la misma?

TP 1er nivel / Principal			Tp 2do nivel - Pág 3			Tp 2do nivel - Pag 5			TLB(valores en hexa)		
Pag	Marco	P	Pag	Marco	P	Pag	Marco	P	Pag	Marco	Proceso
0	40	1	0	15	0	0	12	0	1020	11	A
1	2	0	1	12	0	1	12	1	A101	A	A
2	2	0	2	2	1	2	2	0	0301	AA	B
3	10	0	3	11	1	3	10	1	1111	10	A
4	30	0	4	1	0	4	1	0	0022	4	C
...			...			...					

(tiene más entradas libres)

Bitmap de marcos libres																	
...	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	..
	1	0	1	1	0	1	1	1	1	1	0	1	1	1	1	1	

- 2) En un sistema operativo se dispone de un filesystem unix con inodos de 2 punteros directos, 1 indirecto simple y un indirecto doble. Se tienen bloques de 512 bytes, los cuales insumen 1 ms de tiempo en ser leídos de disco. La unidad de escritura/lectura son bytes, y el acceso a una zona crítica se realiza manipulando las interrupciones. Se dispone de los siguientes procesos:

P1 (llegada: 0)	P2 (llegada: 2)	P3 (llegada: 3)
fseek(file, 0) //cpu, 1 ms write(file, buffer, 1026) escritura = "ok" // cpu, 1 ms	deshab_interrup // cpu, 1 ms calculo_zona_critica() // cpu, 4 ms hab_interrup // cpu, 1 ms fseek(file, 1560); // cpu, 1 ms read(file, buffer, 1) calculo() // cpu, 1 ms	calculo() // cpu, 8 ms

Realice el diagrama de gantt para los siguientes procesos, indicando todos los cambio de modo y las interrupciones ocurridas. Nota: se sabe que el algoritmo es SJF con desalojo y los punteros son de 4 bytes