

Nº Orden	Apellido y nombre	L.U.	Cantidad de hojas

Organización del Computador 2

Recuperatorio del segundo parcial — 07/12/17

1 (20)	2 (30)	3 (50)	
--------	--------	--------	--

Normas generales

- Numere las hojas entregadas. Complete en la primera hoja la cantidad total de hojas entregadas.
- Entregue esta hoja junto al examen, la misma **no** se incluye en la cantidad total de hojas entregadas.
- Está permitido tener los manuales y los apuntes con las listas de instrucciones en el examen. Está prohibido compartir manuales o apuntes entre alumnos durante el examen.
- Cada ejercicio debe realizarse en hojas separadas y numeradas. Debe identificarse cada hoja con nombre, apellido y LU.
- La devolución de los exámenes corregidos es personal. Los pedidos de revisión se realizarán por escrito, antes de retirar el examen corregido del aula.
- Los parciales tienen tres notas: I (Insuficiente): 0 a 59 pts, A- (Aprobado condicional): 60 a 64 pts y A (Aprobado): 65 a 100 pts. No se puede aprobar con A- ambos parciales. Los recuperatorios tienen dos notas: I: 0 a 64 pts y A: 65 a 100 pts.

Ej. 1. (20 puntos)

Responder detalladamente las siguientes preguntas, ejemplificar de ser posible.

- (6p) 1. ¿Qué es el TLB (Translation Lookaside Buffer)? ¿Para qué sirve?
- (6p) 2. Suponga que el PIC indica que la interrupción de reloj es la 32 y la IDT indica que la interrupción 32 se atiende con la función `_isr32`. ¿Es cierto que `_isr32` se va a ejecutar en cada tick de reloj, es decir, cada vez que el reloj prenda su línea en el PIC? Justifique.
- (4p) 3. ¿Cuántos bytes de tamaño, tiene un segmento de límite 0 y granularidad 1?
- (4p) 4. ¿Cuál es el nivel de privilegio necesario en un descriptor de interrupciones que atiende interrupciones de hardware? ¿Cómo se establece?

Ej. 2. (30 puntos)

Se desea analizar las propiedades matemáticas de un mapeo de páginas dado. En particular, si las funciones que describen son monótonas crecientes o no. Una función $f : A \rightarrow B$ es monótona creciente si y solo si para todo x, x' en X , $x < x' \implies f(x) \leq f(y)$.

En nuestro caso, A son todas las direcciones virtuales mapeadas por el CR3, B las direcciones físicas, y f es el mapeo dado por el CR3. Notar que f está definida en todo su dominio pues solo consideramos las direcciones virtuales para las cuales hay un mapeo.

La signature de la función pedida es `int es_monotona_creciente(unsigned int base_dir);`, donde el resultado es 0 si no es monótona y 1 si lo es.

1. (15p) Escriba el código en C de la función pedida.
2. (10p) Modifique la función anterior para considerar las direcciones virtuales que son accesibles **sólo** como supervisor (es decir, A no son todas las direcciones mapeadas, si no aquellas accesibles sólo como supervisor).
3. (5p) El mapeo de las tareas del TP3, ¿es monótono creciente?

Nota: Considerar que los marcos de página son de 4kb.

Ej. 3. (50 puntos)

Tenemos un sistema en el que hay N tareas corriendo concurrentemente. El kernel se encuentra en las direcciones físicas de la 0x00000000 a la 0x003fffff, y las tareas no pueden leerlo ni escribirlo, pero lo tienen mapeado con privilegio supervisor.

En este sistema hay una syscall llamada **atacar** que le permite a una tarea atacar a otra. La syscall sólo toma como parámetro el ID de la tarea a atacar. La syscall debe provocar que la tarea atacada incurra en una excepción, por ejemplo que desreferencie la dirección 0, que divida por 0, o que ejecute un opcode inválido. La syscall **atacar** **no** modifica el código del kernel, en particular, no modifica el código de la rutina de atención del reloj.

Además, si una tarea provoca una excepción luego de ser atacada, debe ser vuelta a correr desde el principio (sus registros de propósito general y su mapeo de páginas no debe alterarse). Si una tarea provoca una excepción en cualquier otra circunstancia debe ser desalojada.

Sugerencias:

- El opcode de la instrucción `mov [0], eax` es A3 00 00 00 00 00 00 00.
- Puede asumir como dada la función `mapear_pagina`.
- Puede asumir que todas las tareas tienen libre la página virtual 0x00500000.
- Si quiere usar el EIP que está en la TSS de la tarea a atacar, piense bien a donde apunta.

- (12p) 1. Indicar los campos relevantes de todas las estructuras involucradas en el sistema para administrar segmentación, paginación, tareas, interrupciones, privilegios, registros de control y funciones del scheduler. Explicar como deben instanciarse las estructuras de datos y explicar su funcionamiento.
- En caso de hacer asunciones no triviales sobre el scheduler, debe escribirse código C que indique las estructuras y las funciones que asumen que tienen.
- (18p) 2. Programar en ASM/C la rutina de atención de la syscall **atacar**.
- (12p) 3. Programar en ASM/C la rutina de atención de una excepción.
- (8p) 4. Programar en ASM/C la rutina de atención de interrupciones del reloj.