

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

## Organización del Computador 2

### Recuperatorio del segundo parcial — 05/07/18

1 (30)	2 (40)	3 (30)	
--------	--------	--------	--

#### 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. (30 puntos)

- (20p) a. Considerando la siguiente tabla de traducciones de direcciones por segmentación y paginación. Dar un conjunto de descriptores de segmento, directorio de paginas y tablas de paginas que cumplan con todas las traducciones. Detallar todos los campos de estructuras involucrados. Además indicar desde que segmento de código se esta ejecutando cada acceso y si la traducción es *identity mapping*.

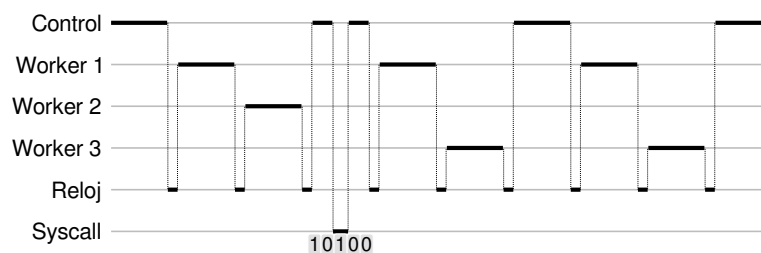
Lógica	Lineal	Física	Características
0x0023:0x00231432	0x09731432	0x83249432	Ejecución como nivel 3 a nivel 0, segmento conforming.
0x0041:0x21032414	0x21032414	0x83A49414	Lectura como nivel 3 a nivel 3
0x0061:0x88493253	0x88494000	0xAA494000	Escritura como nivel 0 a nivel 0
0x0072:0xAAAAFF00	0xAAAAFF00	0xAAAAFF00	Lectura como nivel 0 sobre nivel 3, espacio de solo lectura.

- (10p) b. Diseñar un mecanismo para, desde nivel 0, leer un byte desde una posición física de memoria cualquiera. Suponer que se tiene segmentación y paginación activa, pero no se puede suponer nada de su funcionamiento.

### Ej. 2. (40 puntos)

Considerar un sistema en dos niveles de protección, que ejecuta concurrentemente 6 tareas. Una de estas denominada *control* y las restantes cinco *workers*. La tarea control es ejecutada concurrentemente todo el tiempo, mientras que las tareas workers pueden estar siendo ejecutadas o no.

El sistema posee un servicio que solamente puede ser utilizado por la tarea control, y permite indicar si alguna tarea worker debe ser ejecutada o no. El servicio del sistema atiende en la interrupción 0x62, esta toma como parámetro en *eax* un puntero a un vector de bytes de 5 posiciones, que indica con un 1 si la tarea en esa posición debe ser ejecutada o 0 si la tarea no debe ser ejecutada.



En el ejemplo se observa que antes de llamar al servicio, el sistema se encuentran ejecutando la tarea 1 y 2. Luego de llamar, se continua ejecutando la tarea 1, se deja de ejecutar la tarea 2 y se comienza a ejecutar la tarea 3.

Además, las tareas workers pueden producir excepciones que deben ser capturadas. Dichas tareas deben ser marcadas para que jamás vuelvan a ejecutar, independientemente de lo indicando por la tarea control.

La organización de la memoria debe respetar las siguientes restricciones:

- Los códigos y datos de todas las tareas son independientes y ocupan 1Mb.
- Las tareas workers deben tener, además de su código y datos, un área adicional de datos de 1Mb.
- La tarea control debe tener mapeadas todas las áreas adicionales de todos los workers para poder leer o escribir.

- (10p) a. Describir los campos relevantes de todas las estructuras involucradas en el sistema. Detallar el esquema de segmentación y paginación utilizado. Indicar las estructuras para administrar tareas y como deben ser inicializadas. Indicar como completar las tablas de interrupciones. Considerar en cada caso privilegios involucrados.
- (10p) b. Escribir en ASM/C la rutina de atención de interrupciones del Reloj.
- (20p) c. Escribir en ASM/C una rutina de atención de excepciones y la rutina de la syscall 0x62.

Nota: Suponer que la tarea control no produce ninguna excepción.

### Ej. 3. (30 puntos)

Sea un sistema con segmentación y paginación activa, que ejecuta concurrentemente  $n$  tareas. Se desea implementar un servicio del sistema denominado **topeDePila**, que dado un identificador de una tarea, obtiene el contenido del tope de la pila de nivel 3 de otra tarea.

**topeDePila** toma por parámetro en **eax** el identificador de la tarea y retorna en el mismo registro el valor de 4 bytes contenido en el tope de la pila de nivel de usuario de la tarea indicada.

Suponer que se tiene además la función: `tss* getTssFromId(int id)`, dado un `id`, retorna un puntero a la tss de la tarea.

- (10p) a. Explicar el funcionamiento del servicio pedido y las restricciones que dicho sistema debe tener. Considerar que las pilas de nivel cero de todas las tareas **no están mapeadas con identity mapping**.
- (20p) b. Implementar en ASM/C la rutina del servicio **topeDePila**.