



Apellido y Nombres	Legajo	Calificación

**Parte Teórica: Tiempo Límite 45 minutos**

- 1) Mediante una puerta de tarea se conmuta de una tarea de nivel 0 a otra de nivel 3 quedando el selector DS apuntando al descriptor 5 de la GDT cuyo DPL es 3; y el selector de código CS apuntando al descriptor 6 de la GDT.
  - a. Con qué valor quedan los registros CS y DS?
  - b. Con qué RPL queda el selector DS?
  - c. Explique el mecanismo del proceso y justifique su respuesta a los ítem a y b.
  - d. ¿Como queda el flag NT de la tarea de nivel 0 si ésta se activo por interrupción de hardware? ¿En que estado quedó la tarea que fue interrumpida por ésta?
  
- 2) Un procesador IA-32 tiene el bit PG del registro CR0 en 1. En tales condiciones, un programa ejecuta la instrucción  
`mov eax, [buffer]`  
La dirección lógica de buffer es 0x23:0x60000000. En la gdt el 5to. descriptor tiene los siguientes valores: Base=0x00000000, Límite = 0xFFFFF, D/B=1, G=1, AVL=0, P=1, DPL=11, S=1, Tipo= 010, A=1. El DPT comienza en 0x11000, y a continuación están las 4096 tablas de páginas que se requieren.
  - a. Que número de descriptor del DPT corresponde a la dirección lineal resultante de la instrucción anterior?
  - b. Que valor se debe colocar en ese descriptor del DPT?
  - c. Que valor se debe colocar en el registro CR3?
  - d. Que valor debe ir en el descriptor de la tabla de página, apuntado por el descriptor del DPT especificado en b) para que se acceda a la dirección física 0x0010.0000
  - e. Que ocurre si en el descriptor del punto d. el bit menos significativo se encuentra en "1"?
  - f. Detalle la operatoria a seguir y en que bloque de programa la ubicaría para resolver la situación explicada en e.
  
- 3) Modelo SIMD
  - a. Explique su principio de funcionamiento.
  - b. Implementación de este modelo en los procesadores IA-32: diferentes arquitecturas que se fueron introduciendo, registros y tipos de datos asociados.



Apellido y Nombres	Legajo	Calificación

**Parte Práctica: Tiempo Límite 1 hora 30 minutos**

Se desea implementar un sistema de aula virtual. Para ello se dispone de un servidor con un conjunto de aplicaciones que funcionan enviando en tiempo real una ventana "Blackboard" (a medida que esta se actualice) a los clientes de la red que están asistiendo a la clase virtual a fin de poder visualizar el contenido que va siendo modificado por el presentador.

Ud. debe desarrollar un programa que:

- 1) Al ser instanciado genere un proceso hijo y remplace al mismo por el programa /usr/virtualblackboard/frontend
- 2) El proceso hijo creado en 1) y cuyo código fue remplazado funcionara como una ventana gráfica interactiva con la cual opera el presentador. Cada vez que el contenido del blackboard sea actualizado, la aplicación frontend enviará una señal al pid correspondiente al proceso padre que es quien ejecuta el programa que ud. debe desarrollar.
- 3) El programa a desarrollar, luego de crear al proceso hijo, debe funcionar como servidor concurrente TCP escuchando conexiones por el puerto 10540 y generar un child por cada conexión establecida para luego enviar las muestras del blackboard a los clientes por el puerto UDP 16389. Las muestras tienen un formato de escala de grises de 8bit en una matriz de 640x480 y se obtienen de una shared memory. (ud es libre de elegir el formato de la shared memory, y del sistema de sincronismo entre el "frontend" y el resto de los procesos del servidor)