SO2

Nombre:
DNI:

Primer control de teoría

Todas las respuestas se tienen que justificar brevemente.
Una respuesta sin justificar se interpretará como no contestada.

1. *(4 puntos) Preguntas cortas*

a) ¿Qué es la TSS? ¿Cuándo se utiliza?

b) ¿ Qué guarda Intel automáticamente en la pila al producirse una interrupción y cambiar el nivel de privilegio? Dibuja el contenido de esta pila.

c) ¿Qué diferencia hay entre la tabla de páginas y la TLB?

d) ¿Qué es la IDT y qué contiene?

e) ¿Para que usa Intel el registro hardware CR3?

Nombre:
DNI:

f) ¿Por qué en la llamada a sistema *fork* hay que construir un contexto especial en la pila para el proceso hijo?

g) Indica para las siguientes direcciones su pàgina lógica correspondiente
   i. 0xFFF.....
   ii. 16042.....
   iii. 0xDEAD.....
   iv. 0xCACA.....

h) En un sistema tipo ZeOS con planificación Round Robin, ¿es posible que se entre en modo sistema a traves del handler de una llamada a sistema y se vuelva a modo usuario a través del handler de una interrupción?

i) En un sistema con una política de planificación Random con el siguiente grafo de estados para sus procesos:

¿Qué situaciones pueden provocar la ejecución de la política de planificación?

j) ¿Cual es la parte que penaliza más el rendimiento en el cambio de contexto?

2. *(5 puntos) Gestión de procesos*

Queremos añadir una nueva llamada a sistema a ZeOS que nos permita consultar el estado de los hijos de un proceso. Un proceso sólo puede estar en 2 estados: vivo o muerto.

Queremos añadir las siguientes llamadas a sistema:

int num_hijos(void)

Devuelve el número total de hijos vivos del proceso actual.

int is_alive(int pid)

Devuelve 1 si el proceso pid hijo del proceso actual está vivo o 0 en caso contrario.

Contesta a las siguientes preguntas **justificando tus respuestas** (cualquier respuesta sin justificación se considerará errónea).

a) (*0,75 puntos*) Indica cuáles de los siguientes componentes de ZeOS será necesario modificar para adaptarlos a esta nueva funcionalidad.

Componente	Se modifica/no se modifica	¿Cómo se modifica?/¿Por qué no se modifica?
PCB		
sys_fork		
exit (wrapper)		

Nombre:
DNI:

b) (*0,5 puntos*) ¿Habría que modificar el planificador de procesos? En caso afirmativo describe los cambios necesarios.

c) (*1 punto*) ¿Que tienes que modificar en el sys_exit? Justifica tu respuesta.

d) (*1 punto*) ¿Qué tiene que hacer el código del *sys_num_hijos*()?

e) (*1 punto*)¿Qué tiene que hacer el código del *sys_is_alive(int pid*)?

f) (*0,75 puntos*) Con esta nueva funcionalidad, programa una función en modo usuario que permita a un proceso detener su ejecución hasta la finalización de todos sus hijos.

Nombre:
DNI:

3. *(1 punto) Espacio de direcciones*

Suponiendo un sistema operativo tipo ZeOS en el que toda la memoria física está inicialmente inicializada a 0, que la TLB está vacia, que el espacio de direcciones de los procesos es el típico que hemos visto en clase y donde A, B y C son variables de tipo entero situadas en las direcciones lógicas 0x108008, 0x109008 y 0x110008 respectivamente. Un proceso de usuario ejecuta el siguiente fragmento de código:

```
A = 42;
magical_function();
printf("A = %d\n", A);
printf("B = %d\n", B);
printf("C = %d\n", C);
```

Y como resultado muestra la siguiente salida (la rutina *printf* es la típica rutina para visualizar cadenas de carácteres por la consola):

```
A=42
B=42
C=42
```

Sabiendo que la rutina *magical_function* solo hace una llamada a sistema a una rutina que no toca en ningún momento las variables A, B o C (ni sus direcciones asociadas)... ¿que hace internamente esta rutina de sistema? Indica el código mínimo de esta rutina para obtener este resultado.

void set_ss_page(int logic, int frame) : Modifica la tabla de páginas activa para mapear la página lógica *logic* en el frame físico *frame* .
void del_ss_page(int logic): Elimina el mapeo de la página lógica *logic.*
int get_frame(int logic): Retorna el frame asociado a la página lógica *logic.*
void free_frame(int frame): Marca el frame *frame* como libre.
void set_cr3(): Elimina las entradas de la TLB.