

Examen de Sistemas Operativos

Junio 2008

(2 puntos por cuestión)

- 1) Escribir una función que sea capaz de ejecutar los dos programas cuyos nombres se le pasan como argumentos, y que pinte un mensaje de aviso cuando los dos programas hayan finalizado. La especificación de la función es la siguiente:

```
void ejecuta (const char *prog1, const char *prog2);
```

- 2) Escribir un programa que borre del directorio de trabajo todos los ficheros que se le pasen como argumentos. Si el programa se llama `borra`, el siguiente ejemplo de llamada borraría los ficheros `uno.txt` y `a.out`:

```
[xxxxx]$ borra uno.txt a.out
```

- 3) La función `recurso_compartido` permite el acceso a un recurso compartido que puede ser utilizado simultáneamente hasta por 5 procesos, aunque la propia función no tiene ningún tipo de protección frente a los accesos simultáneos. Proponer el uso de un mecanismo de sincronización para la llamada a esta función que le proporcione la protección requerida, y escribir el código correspondiente. La función se encuentra definida en el fichero de cabeceras `recursos_compartidos.h` y presenta la siguiente especificación:

```
void recurso_compartido ();
```

- 4) Escribir un fragmento de programa que espere con `nanosleep()` hasta que transcurran 1,435 segundos. En caso de que la función retorne por una señal escribir en la pantalla los segundos que realmente se ha esperado.
- 5) La instrucción `"gnatmake -gnatc prueba.adb"` chequea la sintaxis del fichero `prueba.adb` que contiene código en lenguaje Ada. Crear un *script* para una *shell* de Unix que permita chequear la sintaxis Ada de todos los ficheros terminados en `".adb"` del directorio que se pasa como primer parámetro, y mover al directorio que se pasa como segundo parámetro todos aquellos que hayan tenido errores. Realizar el tratamiento de errores. Ejemplo de llamada al *script* "chequea_Ada":

```
[xxxxx]$ chequea_Ada dir_fuentes dir_erroneos
```