

Memoria práctica 2

Carlos Gómez-Lobo Hernaiz, Rodrigo Lardiés Guillén

Ejercicio 2. Escribe un programa en C, ejercicio2.c, que creará 4 pro-cesos hijos en un bucle de forma paralela. Cada hijo, imprimirá el mensaje "Soy el proceso hijo<PID>", después dormirá 30 segundos, imprimirá el mensaje "Soy el proceso hijo <PID> y yame toca terminar.". Tras imprimir este mensaje, el hijo finalizará su ejecución. El padre, tras crear un hijo, dormirá 5 segundos y después enviará la señal SIGTERM al hijo que acaba de crear. Acontinuación, creará el siguiente hijo. ¿Qué mensajes imprime cada hijo? ¿Por qué?

-La salida del programa ha sido:

```
Soy el proceso hijo 5681
Soy el proceso hijo 5684
Soy el proceso hijo 5685
Soy el proceso hijo 5686
```

Esto se debe a que, como cada hijo se va a dormir 30 segundos después de enviar el primer mensaje, 5 segundos después el padre envía la señal SIGTERM y no tiene ningún tratamiento de señales; el proceso hijo es finalizado antes de poder imprimir el segundo mensaje.

Ejercicio 3. Dado el código de ejemplo anterior, contesta a las si-guientes cuestiones:

a) ¿La llamada a sigaction supone que se ejecute la función manejador?

-No, solo establece que si el proceso recibe la señal SIGINT la función manejador es al que ha de ejecutarse, pero si esta no llega, el código se ejecutará normalmente.

b) ¿Cuándo aparece el printf en la pantalla?

-Cuando se pulsa la combinación de teclas Ctrl + C, que hace que se mande la señal SIGINT al proceso, provocando así que se ejecute la función manejador que contiene el printf.

c) ¿Qué ocurre por defecto cuando un programa recibe una señal y no la tiene capturada?

-Cuando a un proceso le llega una señal y no la tiene capturada, el núcleo llama al manejador de dicha señal por defecto, que suele finalizar el proceso y en algunos casos generar un fichero "core".

d) Modifica el programa anterior en un nuevo ejercicio3d.c que capture la señal SIGKILL y cuya función manejadora escriba "He conseguido capturar SIGKILL". ¿Por qué nunca sale por pantalla "He conseguido capturar SIGKILL"?

-No aparece el mensaje porque nunca llega siquiera a mandarse la señal, dado que al llamar a la función sigaction para establecer la función “manejador” como manejador de la señal SIGKILL, aparece un error dado que la señal SIGKILL, al igual que la SIGSTOP no son bloqueables.

Ejercicio 6. ¿Qué sucede cuando el hijo recibe la señal de alarma?

-Que espera a terminal el bucle, y después cuando quita la señal SIGALRM de la máscara de señales, el manejador por defecto la atiende, finalizando el proceso.

Ejercicio 8.

a) ¿Qué pasa cuando SECS=0 y N_READ=1? ¿Se producen lecturas? ¿Se producen escrituras? ¿Por qué?

- Se producen tanto lecturas como escrituras. Se producen aproximadamente el mismo numero de lecturas que de escrituras debido a que solo hay un proceso que lee y otro que escribe. El orden depende de la velocidad relativa de los procesos. Cuando se abre una lectura o escritura, esta se cierra antes de empezar la siguiente acción.

b) ¿Qué pasa cuando SECS=1 y N_READ=10? ¿Se producen lecturas? ¿Se producen escrituras? ¿Por qué?

-Se producen tanto lecturas como escrituras. Ahora se producirán muchas más lecturas que escrituras. Hay 10 procesos para leer y tienen preferencia respecto a la escritura. Se suelen realizar 10 lecturas antes de una escritura, ya que hay un segundo de espera. Pero no tiene porque ser siempre así ya que depende de la velocidad relativa de los procesos.

c) ¿Qué pasa cuando SECS=0 y N_READ=10? ¿Se producen lecturas? ¿Se producen escrituras? ¿Por qué?

- Ahora se producen muchísimas menos escrituras, ya que al no haber tiempo de espera, todos los procesos van a la vez y como hay 10 procesos de lectura (hijos) frente a 1 de escritura (padre), estás son mucho mas escasas

d) ¿Qué pasa si los procesos escritores/lectores no duermen nada entre escrituras/lecturas (si se elimina totalmente el sleep del bucle)? ¿Se producen lecturas? ¿Se producen escrituras? ¿Por qué?

- Si eliminamos el sleep tanto en escritura como en lectura, vuelve a ocurrir lo mismo que en el apartado anterior. Solo que ahora además, después de la primera escritura, esta tendrá que esperar mientras que los hijos no, siendo así las lecturas mucho más abundantes.