SISTEMAS OPERATIVOS II

Tercer curso Ingeniería Informática. Curso 2006-2007

Práctica 4: Procesos en UNIX: señales

Continuar la codificación de un intérprete de comandos (shell) en UNIX. Al igual que en la práctica anterior

- Los argumentos entre corchetes [] son opcionales.
- Los argumentos separados por `|` indican que debe ir uno u otro, pero no ambos simultaneamente.
- No debe dilapidar memoria (ejemplo: variable que se asigna cada vez que se llama a una función y no se libera).
- Cuando el shell no pueda ejecutar una acción por algún motivo, debe indicarlo con un mensaje como el que se obtiene con `sys_errlist[errno]` o con *perror()* (por ejemplo, si no puede cambiar de directorio debe indicar por qué).
- En ningún caso debe producir un error de ejecución (segmentation, bus error ...), salvo que se diga explicitamente
- Las direcciones de memoria deben mostrase en **hexadecimal**.
- La información que se nuestra en pantalla no debe incluir en ningun caso líneas en blanco.
- El shell leerá de su entrada estándar y escribirá en su salida estándar, de manera que podría ser ejecutado un archivo de comandos invocando al shell con su entrada estándar redireccionada a dicho archivo.

En esta práctica el shell incorporará las funciones para comprobar el funcionamiento de los manejadores de señales. Todo el manejo de señales se hara con la interface de System V R4 (*sigaction, sigprocmask ...*). Los nombres de la señales se especificarán sin el SIG. Pueden usarse las funciones que se suministran para convertir entre nombre de una senal y su valor como entero.

pila [***tam***] [***addr***] Establece una pila alternativa para la ejecución de los manejadores de las señales (los que se instalen con el flag SA_ONSTACK). *tam* representa el tamaño y *addr* la dirección. Si se especifica un solo argumento se entiende que es el tamaño y la dirección se obtendrá asignándola con *malloc()*. Si no se especifica ni tamaño ni dirección nos informará del tamaño de la pila alternativa y de su dirección.

senal [S1] [S2] . . . No da información del estado de cada señal: manejada (con la dirección del menjador, los flags y la máscara asociada), ignorada o acción por defecto, asi como de si está enmascarada o no. Ejemplo

```
#senal INT HUP SEGV
INT  Enmacarada     manejador 0x30045a00 SA_RESTART SA_NODEFER. mascara asociada
HUP  No enmascarada Accion por defecto
SEGV No emmascarada Ignorada
```

Si no se especifican señales nos informa de todas

mask [-n] [S1] [S2] . . . Enmascara (mediante *sigprocmask*) las señales S1 S2 . . . .Con -n las desenmascara. Si no se especifican señales nos informa de como están las señales (igual que el comando senal sin argumentos).

ignorar [S1] [S2] . . . Ignora las señales S1 S2 . . . . Si no se especifican señales nos informa de como están las señales (igual que el comando senal sin argumentos).

defecto [S1] [S2] . . . Pone las señales S1 S2 . . . a su acción por defecto. Si no se especifican señales de como están las señales (igual que el comando senal sin argumentos).

conta [-r] [S1] [S2] . . . Nos informa de los contadores de la señales S1 S2 . . . , con -r resetea los contadores de dichas sñales. Si no se especifican señales lo hace para todas (tanto mostrar los contadores como resetearlos en su caso)

handler[-v] [-r] [-p] [-t] [-f] [-dN] [-mSEN1] [-mSEN2] . . . SIG [SIG2] . . . .
Instala un manejador para las señales SIG1, (SIG2 . . .) (Recordar que el manejador es la función que se ejecuta CUANDO llega la señal). El manejador incrementará un contador que indica cuantas veces se ha ejecutado. Además se admitirán las siguientes opciones

- -mSEN Se añade SEN al conjunto de señales enmascaradas DURANTE la ejecución del manejador (miempro sa_mask de la estructura *sigaction*)
- -dN El manejador se queda en espera N segundos (con la llamada *sleep*). Dicha espera debe ser la última instrucción dentro del manejador
- -v El manejador al ejecutarse debe imprimir en pantalla la señal que se ha recibido, cuantas veces se ha ejecutado dicho manejador (el valor del contador) así como la dirección del parámetro que recibe. De no indicarse -v el manejador NO DEBE IMPRIMIR NADA en pantalla.

-r El manejador reenvía al proceso la señal para la cual es manejador. En caso de reenviarse la señal, debe hacerse después de imprimir en pantalla (en caso de que se imprima) y antes de quedarse en espera (en caso de que se haya especificado -sN)

-t El manejador se instala con el flag SA_RESETHAND

-p El manejador se ejecutará en la pila alternativa (se instala con el flag SA_ONSTACK

-f El manejador se instala con el flag SA_NODEFER

maxhandler [*n*] Establece el número máximo de veces que el manejador instalado con *-r* se reenvía la señal. Si no se especifica n se indicará a cuanto está establecido dicho valor. Un valor 0 indica que no ha límite predefinido.

bucle Hace que el shell entre en un bucle infinito. Instala un manejador para SIGINT que permite salir del bucle pulsando control-c para seguir ejecutando el shell.

segmentation produce un fallo de segmentación en el shell. (No vale enviar SIGSEGV, tiene que ser un fallo de segmentación de verdad).

fpe produce una excepción de la unidad en punto flotante en el shell. (No vale enviar SIGFPE, tiene que ser una excepción de la unidad en punto flotante de verdad).

Información detallada de las llamadas al sistema y las funciones de la librería debe obternerse con man (sigaction, sigprocmask, sigaltstack, . . .)

FORMA DE ENTREGA Como en prácticas anteriores

FECHA DE ENTREGA Viernes 15 Junio 2007