



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA DE INGENIERÍA Y
CIENCIAS SOCIALES Y ADMINISTRATIVAS



Práctica 7: Señales

Yong Rodríguez Luz María

Cuando queremos transmitir información corremos el riesgo de que la información no llegue a su destino y el proceso que esperaba la llegada de esa información se quede bloqueado. Se deben implementar mecanismos para poder salir del bloqueo y las señales son la solución a esta situación.

Una señal es una interrupción que envía el kernel hacia un proceso. Esta interrupción puede ser originada por una combinación de teclas, ejecutando un comando, a través de un proceso o a través de un cronometro.

Ejercicios prácticos.

Pregunta 7-1: ¿Cuántas señales están definidas en su sistema?

36

¿Cuántas señales solo detienen el proceso?

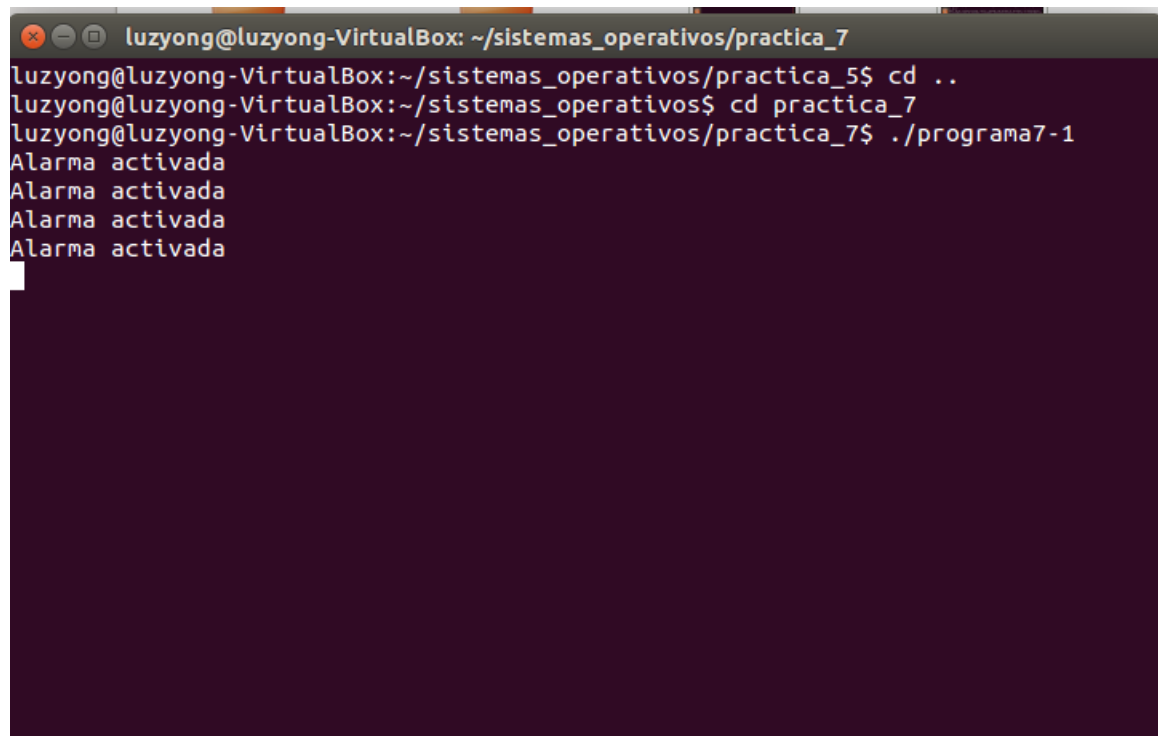
3

¿Cuáles señales no pueden ser capturadas y provocan irremediablemente que un proceso muera?

2, SIGKILL y SIGSTOP

Programa 7-1.

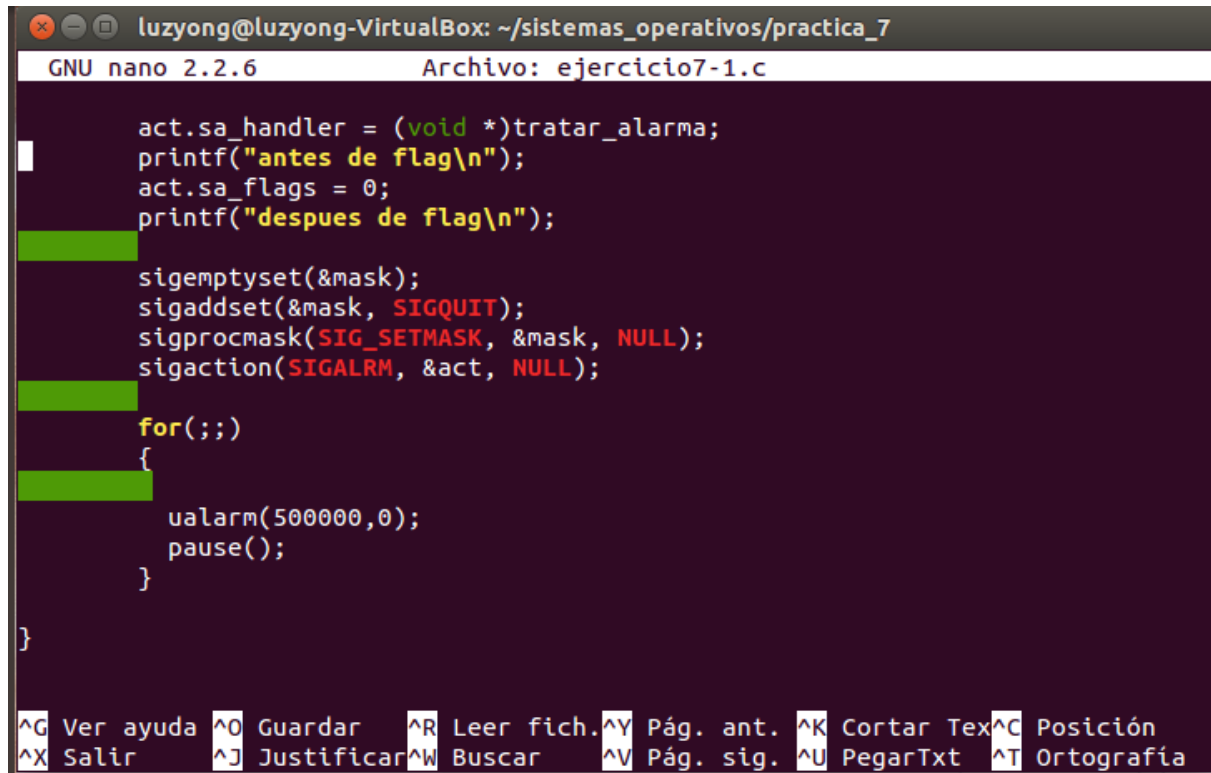
Ejecuta una función cuando llega una señal activada por un cronómetro. En este caso es la señal SIGALRM. Cada 3 segundos se ejecuta la función `tratar_alarma()`.



```
luzyong@luzyong-VirtualBox: ~/sistemas_operativos/practica_7
luzyong@luzyong-VirtualBox:~/sistemas_operativos/practica_5$ cd ..
luzyong@luzyong-VirtualBox:~/sistemas_operativos$ cd practica_7
luzyong@luzyong-VirtualBox:~/sistemas_operativos/practica_7$ ./programa7-1
Alarma activada
Alarma activada
Alarma activada
Alarma activada
```

Ejercicio 7-1

Modifique el programa para que se envíe la señal SIGALRM cada medio segundo. Modifique la función manejadora para que imprima el número de señal que le envía el kernel a nuestro proceso.



```
luzyong@luzyong-VirtualBox: ~/sistemas_operativos/practica_7
GNU nano 2.2.6 Archivo: ejercicio7-1.c

    act.sa_handler = (void *)tratar_alarma;
    printf("antes de flag\n");
    act.sa_flags = 0;
    printf("despues de flag\n");

    sigemptyset(&mask);
    sigaddset(&mask, SIGQUIT);
    sigprocmask(SIG_SETMASK, &mask, NULL);
    sigaction(SIGALRM, &act, NULL);

    for(;;)
    {
        ualarm(500000,0);
        pause();
    }
}
```

Ver ayuda Guardar Leer fich. Pág. ant. Cortar Text Posición
Salir Justificar Buscar Pág. sig. PegarTxt Ortografía

Pregunta 7-2: Envíe la señal SIGALRM desde otra terminal con el comando kill y explique lo que pasa. ¿Qué sucede si añade SIGALARM a la máscara?

Si se añade SIGALARM a la máscara, ignora esa señal y ya no se ejecutaría la impresión del mensaje cada 3 segundos.

Pregunta 7-3: ¿Cuál es la línea de código en el programa 7-1 que se ejecuta inmediatamente después de ejecutarse el manejador de señal?

La siguiente línea que ejecuta el programa después del manejador es la del for y de ahí se va directamente a la función tratar_alarma.

Ejercicio 7-2

Modifique el programa para que la señal SIGQUIT (CTRL-\) deje de ser ignorada y provoque su terminación.


```

luzyong@luzyong-VirtualBox: ~/sistemas_operativos/practica_7
luzyong@luzyong-VirtualBox:~/sistemas_operativos/practica_7$ ./programa7-2
Padre: Nomas 5 segundos, eh
Hijo: Rápido, mándame un SIGUSR1 a este pid:3207
^C
luzyong@luzyong-VirtualBox:~/sistemas_operativos/practica_7$ ./ejercicio7-2
antes de ir a tratar alarma
antes de flag
despues de flag
Alarma activada
numero de señal:14 pid:3208
Alarma activada
numero de señal:14 pid:3208
Alarma activada
numero de señal:14 pid:3208
Alarma activada
numero de señal:14 pid:3208
Alarma activada
numero de señal:14 pid:3208
^\\Abandona ('core' generado)
luzyong@luzyong-VirtualBox:~/sistemas_operativos/practica_7$ █

```

Ejercicio 7-3

Modifique el programa para evitar que se detenga con la señal 18 SIGTSTP y que además imprima el mensaje “Me han intentado detener”.



The image shows two terminal windows. The main terminal window on the left has a dark purple background and displays a repeating sequence of messages: 'numero de señal:14 pid:4925' followed by 'Alarma activada'. The sequence is interrupted by a prompt 'Me han intentado detener >:' and a command '>:c'. A second terminal window, titled 'luzyong@luzyong-VirtualBox: ~', is overlaid on the right. It shows the execution of 'kill -tstp 4896' and 'kill -cont 4925' commands, followed by a new prompt.

```
luzyong@luzyong-VirtualBox: ~/sistemas_operativos/practica_7
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
Me han intentado detener >:
>:c

Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925
Alarma activada
numero de señal:14 pid:4925

```

```
luzyong@luzyong-VirtualBox: ~
luzyong@luzyong-VirtualBox:~$ kill -tstp 4896
luzyong@luzyong-VirtualBox:~$ kill -cont 4925
luzyong@luzyong-VirtualBox:~$
```

Modifique el programa para que ninguna señal lo pueda detener.

Ejercicio 7.5

```

luzyong@luzyong-VirtualBox: ~/sisistemas_operativos/practica_7
luzyong@luzyong-VirtualBox:~$ cd sistemas_operativos/practica_7
luzyong@luzyong-VirtualBox:~/sisistemas_operativos/practica_7$ ./programa7-2
Padre: Nomas 5 segundos, eh
Hijo: Rápido, mándame un SIGUSR1 a este pid:3077
Terminado (killed)
luzyong@luzyong-VirtualBox:~/sisistemas_operativos/practica_7$ ./programa7-2
Padre: Nomas 5 segundos, eh
Hijo: Rápido, mándame un SIGUSR1 a este pid:3108
Terminado (killed)
luzyong@luzyong-VirtualBox:~/sisistemas_operativos/practica_7$ ./programa7-2
Padre: Nomas 5 segundos, eh
Hijo: Rápido, mándame un SIGUSR1 a este pid:311
Terminado (killed)
luzyong@luzyong-VirtualBox:~/sisistemas_operativo
Padre: Nomas 5 segundos, eh
Hijo: Rápido, mándame un SIGUSR1 a este pid:311
Terminado (killed)
luzyong@luzyong-VirtualBox:~/sisistemas_operativo
Padre: Nomas 5 segundos, eh
Hijo: Rápido, mándame un SIGUSR1 a este pid:311
Hijo: Me mandaste un SIGUSR1
Hijo: Terminé sin morir en el intento :)
Terminado (killed)
luzyong@luzyong-VirtualBox:~/sisistemas_operativo

```

En mi código le puse más de 5 segundos para que pudiera visualizarse y un tiempo en la función aleatorio para que pudiera pasar que a veces sí alcanza a terminar y a veces no.

Conclusiones.

En esta práctica aprendí a tratar las señales. Se me ocurrieron varias aplicaciones de estas señales que podrían funcionar en el desarrollo de algún programa. Investigué un poco más y utilicé funciones como signal para tratar una señal y con eso poder resolver y experimentar en algunos de los ejercicios que se plantearon en la práctica.