laSalle
UNIVERSITAT RAMON LLULL

Operating Systems 2021-2022

Session 1: Signals 23/09/21

Goals

The objective of this session is to introduce to the student a new tool from the operating system: the signals. This tool is used to generated asynchronous interruptions in one or more processes. With the right knowledge and understanding of the signals students can modify the normal behavior of their programs.

From this session on, the *file descriptors* will be used as the input and output mechanism.

Motivation

More specifically, with this session. The students must practice:

- Capturing and handling signals (*Signal handler*)
- Sending signals (*Kill* and *raise*)
- Periodic signals (*alarm*)
- Other related functionalities (*pause*, *getpid*, etc.)

Previous documentation

To complete this session, it is recommended to read of the following references:

SALVADOR, J. (2009). *Introducció al llenguatge de programació C*.

STEVENS, W. R. & RAGO S.A. (2008). *Advanced Programming in the UNIX Environment*, 2nd edition.

TANENBAUM, A. S. (2009). *Modern Operating Systems*, 3rd Edition.

Chrono-Signal

The time management when implementing the sessions of the Operating Systems subject can be sometimes complicated. The students, knowing this problem, have proposed to implement a program that helps them not forget to deliver the activity on time, notifying them when the class time is near to an ending.

The application to program will act as a countdown chronometer that when it reaches 0, it will notify the student that it's time to deliver the session.

For that, the application will work with the following functionalities and its corresponding shortcuts (signals):

SIGCONT	Start countdown.
SIGABRT	Stop countdown.
SIGSYS	Set time
SIGBUS	Add 5 minutes to the time.
SIGPIPE	Reduce 5 minutes the time.
SIGILL	Restart the timer.
SIGFPE	See the remaining time.
SIGALRM	Time has ended

When the program starts or when the SIGSYS signal is activated, the program will ask for the quantity of time it has to countdown to with the format being HH:MM:SS. When the time is entered, it will start counting.

Finally, when the student decides to close the program, the signal used to close it is the **SIGINT**.

Some of the functionalities listed before can't be executed if other ones haven't been activated previously. These cases will show a message asking for the other functionality to be activated first. For example, stopping the timer needs the timer to be running and vice versa, and modifying the timer when is running or stopped.

Execution Examples

```
> ./S1
SIGSYS -> Enter session time:
      02:30:00
      Countdown of 2h 30m 0s started.
SIGABRT -> Timer paused.
SIGCONT -> Timer resumed.
SIGBUS -> Added 5 minutes to the countdown.
SIGPIPE -> Reduced 5 minutes to the countdown.
SIGILL -> The timer will restart.
      Countdown of 2h 30m 0s started.
SIGSYS -> Enter session time:
      00:01:00
      Countdown of 0h 1m 0s started.
SIGFPE -> 0h 0m 37s remaining.
(SIGALARM) The time has ended, deliver the session.
SIGINT -> Closing program.
Killed
```

```
> ./S1
SIGSYS -> Enter session time:
      02:30:00
      Countdown of 2h 30m 0s started.
SIGCONT -> The timer is already running.
SIGABRT -> Timer paused.
SIGABRT -> The timer is already paused.
SIGINT -> Closing program.
Killed
```

Considerations

- The whole program must be implemented using signals. Other communication mechanisms are prohibited.
- When the program ends, all the dynamic memory used must have been freed, and all file descriptors must be closed.
- The program *output* must be **completely identic** at the one shown in the execution examples.
- The use of global variables is not allowed, except the ones necessary for the control of the signals.
- All inputs and outputs must be done using file descriptors. **The use of printf, scanf, FILE*, getchar or similar is NOT allowed.**
- **You must compile using the –Wall and –Wextra flags.**
- **Any deliverable containing warnings or errors will be directly discarded.**
- **At the start of the .c you must include a comment with your logins, names, and surnames.**
- **For submitting the session, you must hand over a file "S1.c", and delivered through eStudy platform.**