

## Signal7 Demonstration

### Files

The files for this demonstration can be found in the rlogin cluster in the directory

```
/web/courses/cs3214/spring2014/butta/examples/signal-demo/signal7
```

The files are `driver.c` `esh-sys-utils.c` `esh-sys-utils.h` `Makefile`  
`receiver.c` `sender.c`

The “make” command by default will create an executable named `driver`. The `driver` creates two child processes and waits for them to terminate, printing a message as each child terminates. One child process, `sender`, sends a sequence of signals to the other child process, the `receiver`.

### Purpose

The purposes of this demonstration are

- to explore the effect of multiple signals of the same type being received close together
- to see how to send signals programmatically

### Steps

1. Examine the code for the `sender` process in `sender.c`. Note how many `SIGUSR1` signals are sent by the sender.
2. Examine the code for the `receiver` process in `receiver.c`. Note the `catch_signals` signal handler and the loop in the main program.
3. Use the `Makefile` to create the executable programs `driver`, `receiver`, and `sender` using the command “make”.
4. At the shell prompt execute the `driver` program. Observe the output produced.
5. Repeat step 4 several times and observe the output produced in each case.

### Questions

Based on your observations, answer these questions.

1. What system call is used to send a signal?
2. What signal causes the `receiver` to end its loop?
3. What does the `receiver` do when a `SIGUSR1` signal is received?
4. Is there anything in the `receiver` that prevents the reception of a `SIGUSR1` signal?
5. Does the `receiver` receive every signal sent by the `sender`? If not, approximately what percentage of them is “missed”?
6. How do you explain what you answered to question 5?