CSC209 Week 9 Notes

Hyungmo Gu

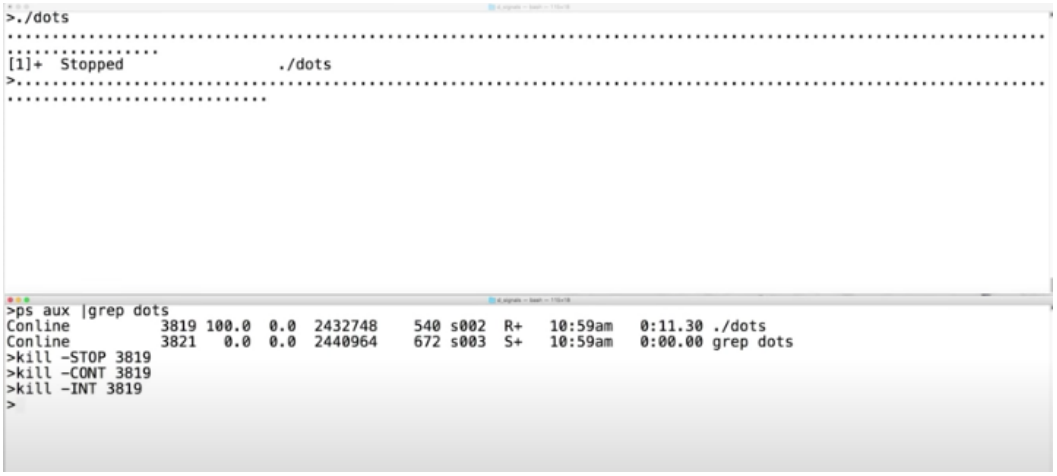
May 16, 2020

Signals 1 of 2

- Introduction to Signals
  - Signals
    - ∗ are mechanisms that allow process or the os to interrupt currently running process and notify that an event has occured

No	Name	Default Action	Description
1	SIGHUP	terminate process	terminal line hangup
2	SIGINT	terminate process	interrupt program
3	SIGQUIT	create core image	quit program
4	SIGILL	create core image	illegal instruction
5	SIGTRAP	create core image	trace trap
6	SIGABRT	create core image	abort program (formerly SIGIOT)
7	SIGEMT	create core image	emulate instruction executed
8	SIGFPE	create core image	floating-point exception
9	SIGKILL	terminate process	kill program
10	SIGBUS	create core image	bus error
11	SIGSEGV	create core image	segmentation violation
12	SIGSYS	create core image	non-existent system call invoked
13	SIGPIPE	terminate process	write on a pipe with no reader
14	SIGALRM	terminate process	real-time timer expired
15	SIGTERM	terminate process	software termination signal
16	SIGURG	discard signal	urgent condition present on socket
17	SIGSTOP	stop process	stop (cannot be caught or ignored)
18	SIGTSTP	stop process	stop signal generated from keyboard
19	SIGCONT	discard signal	continue after stop
20	SIGCHLD		

  - How it Works
    1. Using hotkey
       - ∗ i.e. $CTRL + C$ in terminal sends SIGINT
       - ∗ i.e. $CTRL + Z$ in terminal sends SIGSTOP
    2. Using kill command




```
>>>./signals_example_1.out # <- This is done in separate terminal
>>> ps aux | grep ./signals_example_1.out
>>> kill -STOP <PID>
>>> kill -CONT <PID>
>>> kill -INT <PID>

```