HW 9

Joshua Thompson

March 17, 2018

Questions

1. (5 points) What does it mean that signals arrive asynchronously?

   Means that the portions of the program are executed out of order. Especially when the program is waiting for signals to come from the hardware.

2. (10 points) What signal is generated from the following keyboard-shortcut/command?

   (a) `Ctrl-c`

   SIGINT

   (b) `Ctrl-z`

   SIGSTP

   (c) `fg/bg`

   SIGCNT

   (d) `Ctrl-\`

   SIGKILL

3. (15 points) Run the command kill -1 to list all the signals and their signal numbers. Find either the matching signal-numbers/signal-name for the following values below. Additionally, for each signal, use man 7 signal to describe the default action of each.

   (a) `SIGKILL`

   Number: 9 Action: Term

   (b) `14`

   Signal: SIGALRM Action: Term

   (c) `SIGALRM`

   Number: 14 Action: Term

   (d) `SIGBRT`

Number: 6 Action: Core

(e) `21`

Signal: SIGTTIN Action: Stop

(f) `1`

Signal: SIGHUP Action: Term

(g) `SIGCHILD`

Number: 17 Action: Term

(h) `SIGTTOU`

Number: 22 Action: Stop

4. (15 points) Provide a brief plain-English explanation, e.g., reference which signal is sent and to which processes, for each of the kill/killall commands.

   (a) `killall -17 sleep`

   Terminates all children of sleep processes

   (b) `kill -9 2237`

   Kills process 2237

   (c) `killall -SIGUSR1 a.out`

   Performs a userdefined signal on all a.out processes

   (d) `kill -SIGABRT -1`

   Performs abort signal

   (e) `killall sleep`

   kills all sleep proceses

   (f) `killall -u`

   kills all processes running as USER

5. On a lab machine, run the following program in the background. From the same terminal on the same machine, send the program the following two signals below and describe the results.

   (a) `(3 points) SIGUSR1`

The result is what looks like a cat with a message that says "Ha Ha, missed me!"

(b) (3 points) SIGUSR2

The result is a skull and crossbones with the message "You shot me!"

6. Consider the program below and answer the associated questions:

```
int count = 0;

void handler(int signum){

  printf("You Shot Me!\n");
  count++;

  if(count > 3){
    printf("I'm dead!\n");
    exit(1);
  }

}

int main(){

  //establish signal hander for SIGTINT and SIGSTOP
  signal(SIGTINT,handler);
  signal(SIGTSTP,handler);

  //loop forever
  while(1);

}
```

(a) (3 points) what is the output of the program if the user hits Ctrl-c only once?

Result is "You shot me!"

(b) (3 points) what is the output of the program if the user hits Ctrl-c four times?

```
Result is:
You shot me!
You shot me!
You shot me!
You shot me!
Im dead!
```

(c) (3 points) what is the output of the program if the user hits Ctrl-c three times and Ctrl-z once?

```
Result is:
You shot me!
You shot me!
You shot me!
You shot me!
Im dead!
```

7. (5 points) what does the system call pause() do? Yes, it pauses the program, of course, but until when does the program stay paused and why is it a useful command?

The pause() system call will pause the program until another signal is delieverd and handled. this is useful because it helps remove overhead. Example is an infinite while loop will not continue to run, instead it will be interrupted and only start again when another signal is given.

8. (10 points) How many times does the program below print alarm? Explain.

```
int count = 10;

void handler(int sugnum){

  printf("Alarm!\n");
  count /= 2;
  alarm(count);

}

int main(){

  signal(SIGALRM, handler);
  alarm(count);

  while(1)
  pause();
  }
```

Alarm is printed four times. This is because it is called the first time in the main, then the count happens in the handler. The alarm is then called three more times from looping in the handler function.

9. (10 points) Convert the following use of signal() below to a sigaction() call.

```
signal(SIGUSR1, handler);
```

sigaction(SIGUSR1, handler, NULL);

10. (5 points) what sigaction flag is used to ensure that system calls will be restarted when interrupted?

SA_RESTART

11. (5 points) Provide an example of why the read() system call would need to be restarted due to a signal delivery.

read() system calls need to be restarted because they are not reentrant functions. A reentrant function does not require a signal to continue their function.

12. (5 points) Look in man 7 signal and kill -l and draw a picture of your favorite signal. Be sure to clearly identify it.

Question did not make sense.