

# Advanced Programming

## COMS 3157

April 12, 2025

Patrick Shen

---

1. Q1?

(2 marks)

- (a) What is a signal?
- (b) What is a signal handler?

**Ans:**

- (a) A small message that notifies a process that an event of some type has occurred.
- (b) A signal handler is a function that executes in response to the arrival and consumption of a signal. The signal handler *runs in the process that receives the signal*.

2. Give the scenario where each signal would occur.

(4 marks)

- (a) SIGFPE
- (b) SIGINT
- (c) SIGTSTP
- (d) SIGCONT

**Ans:**

- (a) SIGFPE: Whenever a process commits an integer-divide-by-zero, the kernel signals a **SIGFPE** signal to the offending process.
- (b) SIGINT: When you type ctrl-c, the kernel sends a **SIGINT** to the foreground process (and by default, that foreground is terminated).
- (c) SIGTSTP: When you type ctrl-z, the kernel issues a **SIGTSTP** to the foreground process (and by default, the foreground process is halted until a subsequent **SIGCONT** signal instructs it to continue).

(d) SIGCONT: When a process attempts to publish data to the write end of a pipe after the read end has been closed, the kernel sends a **SIGPIPE** to the offending process.

3. Give the following actions for the predefined signal function handlers in signal() (2 marks)

(a) SIG\_DFL

(b) SIG\_IGN

**Ans:**

(a) SIG\_DFL: clears any custom function handler for signal.  
clears "somehandler" signal function

```
1 int main(void)
2 {
3     ...
4     signal(SIGINT, somehandler);
5     ...
6     signal(SIGINT, SIG_DFL);
7     ...
8 }
9
```

Listing 1: SIG\_DFL example

(b) SIG\_IGN: ignores signals

```
1 int main(void)
2 {
3     ...
4     signal(SIGINT, SIG_IGN);
5     ...
6 }
7
```

Listing 2: SIG\_IGN example

4. Which two signals cannot have any signal handlers? (2 marks)

**Ans:**

(a) SIGKILL (9)

(b) SIGSTOP (19)

5. What does the raise(int iSig) function do?

(1 mark)

**Ans:** Commands OS to send a signal of type iSig to calling process. Returns 0 to indicate success, non-0 to indicate failure.

```
1  iRet = raise(SIGINT);  
2
```

Listing 3: raise() example

raise(SIGINT) sends a 2/SIGINT signal to calling process.

6. What does the kill(pid\_t iPid, int iSig) function do?

(1 mark)

**Ans:** Sends a iSig signal to the process iPid. Equivalent to raise(iSig) when iPid is the id of current process. You must own process pid (or have admin privileges).

```
1  iRet = kill(1234, SIGINT);  
2
```

Listing 4: kill() example

kill(1234, SIGINT) sends a 2/SIGINT signal to process 1234.

7. Q7. [2 marks]

(1 mark)

- (a) What does the alarm(int time) function do?
- (b) What happens if the time argument is set to 0?

**Ans:**

- (a) The alarm function sends the SIGALARM (14) signal, which you can use to catch using the signal function. Below is an example:

```
1  static void myHandler(int iSig)  
2  {  
3      printf("In myHandler with argument %d\n", iSig);  
4      alarm(2); /* Set another alarm */  
5  }  
6  
7  int main(void)  
8  {  
9      signal(SIGALARM, myHandler);  
10     alarm(2); /* Set an alarm */
```

```
11     printf("Entering an infinite loop\n");
12     for (;;)
13         ;
14     return 0;
15 }
16
```

Listing 5: alarm() example

In this code, this would cause an alarm to be set every two seconds, and then the print statement in the signal handler would be printed.

- (b) alarm(0) cancels any pending alarm that has not gone off from any previous alarm() calls.