Tianyi Li
1000929445

CSC209 Week 9 PCRS Practice Questions

```
int val = 10;

void handler(int sig)
{
    val += 5;
}

int main()
{
    signal(SIGCHLD, handler);
    if ((pid = fork()) == 0)
    {
        val -= 3;
        exit(0);
    }
    waitpid(pid, NULL, 0);
    printf("val = %d\n", val);
    exit(0);
}
```

What will happen after running this program?

```
>>> val = 15
The signal handler function “handler” executes after receiving the signal
SIGCHID.
```

```
void handler(int);

void handler(int sig)
{
    printf("CSC\n");
}

int main()
{
    signal(SIGINT, handler);
    while(1){
        printf("209\n");
}
    return 0;
}
```

What happens as SIGINT (Ctrl+C) hits the process while its running?

```
The program keeps printing program keeps printing ‘209’ as it runs, and it
will output “CSC” every time SIGINT hits.
```

Tianyi Li
1000929445

```
void handler(int);

void handler(int sig)
{
    printf("CSC\n");
    signal(SIGINT, SIG_DFL);
}
int main()
{
    signal(SIGINT, handler);
    while(1){
        printf("209\n");
    }
return 0;
}
```

What will happen if we press “Ctrl+c” key two times after running this program?

Process will terminate. According to the “handler” function, as the SIGINT signal arrives two time, the signal performs its default operation (i.e. termination).

```
void handler (int);

void handler (int sig)
{
    printf("CSC\n");
}

int main()
{
    struct sigaction act;
    act.sa_handler = handler;
    act.sa_flags = 0;
    sigemptyset(&act.sa_mask);
    sigaction(SIGINT,&act,0);
    while(1){
        printf("209\n");
}
    return 0;
}
```

What happens as SIGINT (Ctrl+C) hits the process while its running?

The program keeps printing program keeps printing ‘209’ as it runs, and it will output “CSC” every time SIGINT hits.

Tianyi Li
1000929445

```
void handler (int);

void handler (int sig)
{
    printf("%s",sys_siglist[sig]);
}
int main()
{
    signal(SIGINT, handler);
    while(1){
        printf("CSC\n");
    }
    return 0;
}
```

What happens as SIGINT (Ctrl+C) hits the process while its running?

The program keeps printing program keeps printing 'CSC' as it runs, and it will print "Interrupt" every time SIGINT hits. The messages associated with signals can be access by the function sys_siglist().

```
void handler (int);

void handler (int sig)
{
    printf("%s\n",sys_siglist[sig]);
    signal(SIGSEGV,SIG_IGN);
}
int main()
{
    signal (SIGSEGV, handler);
    char *str;
    *str = 10;
    return 0;
}
```

What will happen after running this program?

The segmentation fault occurs because the memory is not allocated to the pointer *str.