Advanced Unix programming – Quiz3 Solutions

1. (1 point) What should the data type of *fds* be, for it to be used in the function call *pipe(fds)*? (**Give a correct declaration of *fds***).

   ```
   int fds[2];
   ```

2. (2 points) Consider the function call: *pipe(fds).* Which of the following **must** be true.

   (a) fds[0] and fds[1] are available for reading.

   (b) fds[0] is available for reading and fds[1] is available for writing.

   (c) fds[0] is available for writing and fds[1] is available for reading.

   (d) fds[0] and fds[1] are available for writing.

   b

3. (1 point) Consider the following situation. A process sets a function called *sig_int* to be the signal handler for SIG_INT. It then execs a program. Will *sig_int* remain the signal handler for SIG_INT in this exec-ed program?

   (a) Yes.

   (b) No.

   (c) All of the above.

   b

   In the rest of the quiz, you may wish to use some of the following functions, data types, and macros (you may need to use a few outside this list too): sigemptyset, sigfillset, sigaddset, sigdelset, sigaction, sigprocmask, sigset_t – *used to store a signal set*, struct sigaction – *which has fields: sa_handler, sa_mask, sa_flags, and sa_sigaction*, SIG_BLOCK, SIG_UNBLOCK, SIG_SETMASK, SIG_DFL, SIG_IGN.

4. (1 points) Write a short piece of code to make the SIG_INT signal ignored. (You may use the function: 'signal', if you wish to.)

   ```
   signal(SIGINT,SIG_IGN);
   ```

5. (4 points) Write a short piece of code to add SIG_INT to the set of signals that are currently blocked. Make sure that you declare correct data types and set up the signal set correctly.

   ```
   sigset_t set;

   sigemptyset(&set);
   sigaddset(&set,SIGINT);
   sigprocmask(SIG_BLOCK, &set, NULL);
   ```

6. (1 point) Write a short piece of code to have SIG_INT handled by a function called sig_int. No additional signal will be blocked during execution of the signal handler, and no special flags will be set. Make sure that you declare correct data types. You may **not** use the 'signal' function.

```
struct sigaction act;

act.sa_handler = sig_int;
sigemptyset(&act.sa_mask);
act.sa_flags = 0;

sigaction(SIGINT, &act, NULL);
```