OS MP1 Handwriting

B08902055 李英華

March 29, 2021

1 Fork-Exec

- We can't see the output message since $exec()$ would not return.
- We can $fork()$ a child process and $exec()$ the program. For the parent, call $wait()$ and then $printf()$ the message.

2 Signal Handled

(3) it depends. The default action of $SIGHUP$ is to terminate the process. If $kill()$ send signal before the signal handler is signed up, then the disposition is termination. On the other hand, if the signal arrived after signing up, then the signal handler would be executed and the message would thus be seen.

3 Fork vs Thread

- Considering the aspect of inter communication. Process only has limited ways to pass data. For example, declaring shared memory by system call or explicitly calling certain system calls to pass messages. However, system calls is expensive.
  On the contrary, resources and memory are shared among threads, and could be accessed through mutexes or conditional variables, for instance, whose overhead is lower.
- Considering the aspect of memory usage, $fork()$ generates the exact copy of current process or Copy-On-Write to get the separate memory image, which results in system overhead. Whereas threads share resources and memory, data exchange between them could be faster and easier.