

HW-8-Project-7-Shell-Part-2

CS 503/571 Systems Basics

100 Points

Summary

Continue to develop and test a small version of a shell. Add functionality to reap terminated (zombie) children.

What To Do

The small shell developed earlier does not reap terminated children. For example: in the following, we start **shellex** and then execute program **myspin**. When **myspin** terminates, it remains a zombie.

```
ck47@tux4:~/cs503$ shellex
> /usr/bin/ps
  PID TTY          TIME CMD
1409104 pts/22    00:00:00 bash
1410642 pts/22    00:00:00 shellex
1410646 pts/22    00:00:00 ps
> myspin 30 &
1410658 myspin 30 &
> /usr/bin/ps
  PID TTY          TIME CMD
1409104 pts/22    00:00:00 bash
1410642 pts/22    00:00:00 shellex
1410658 pts/22    00:00:00 myspin
1410672 pts/22    00:00:00 ps
> /usr/bin/ps
  PID TTY          TIME CMD
1409104 pts/22    00:00:00 bash
1410642 pts/22    00:00:00 shellex
1410658 pts/22    00:00:00 myspin <defunct>
1410687 pts/22    00:00:00 ps
> quit
ck47@tux4:~/cs503$
```

Follow the directions in BLS "How the Shell Works" lecture and BLS "Signals, Pipes, and setjmp" lecture.

Use code in CSAPP2, 8.4.6 "Using fork and execve to Run Programs" and code from CS503-Processes.pdf, slides 54,55, to create a small shell that forks processes in the background and uses SIGCHLD to know when they terminated and reap them.