

Homework

- Implement following operations on your shell.

Command	Description
Execute a user program	Execution command should resemble bash shell. Ex> <code>./hello</code> : execute hello in current directory <code>/bin/ls</code> : execute <code>/bin/ls</code>
process	Report a snapshot of the current process Ref> <code>ps</code>
reap "PID"	Send a kill signal to a process Ref> <code>kill</code> EX> <code>kill 3556</code> : kill the process which has pid 3556
CMD *	Run a background program Ref > <code>CMD &</code> EX > <code>./hello * ></code> run hello in background

Homework

■ Restriction

- No zombie process is allowed

```
proshb@proshb-vm:~/test$ ps
PID TTY          TIME CMD
2686 pts/1        00:00:01 bash
12703 pts/1        00:00:01 zombie
12704 pts/1        00:00:00 zombie <defunct>
12706 pts/1        00:00:00 ps
proshb@proshb-vm:~/test$
```

- Wait for child process

```
int main(void)
{
...
    signal(SIGCHLD, handler);
...
}
```

```
void handler(int sig) {
    pid_t pid;
    int stat;
    while ((pid = waitpid(-1, &stat, WNOHANG)) > 0);
    return;
}
```

ID	Name	Default Action	Corresponding Event
2	SIGINT	Terminate	Interrupt from keyboard (ctrl-c)
9	SIGKILL	Terminate	Kill program (cannot override or ignore)
11	SIGSEGV	Terminate & Dump	Segmentation violation
14	SIGALRM	Terminate	Timer signal
17	SIGCHLD	Ignore	Child stopped or terminated

The only information in a signal is its ID and the fact that it arrived.

Example

- `ps -> kill process`

```
proshb@proshb-vm:~/test$ ps
  PID TTY          TIME CMD
 2686 pts/1        00:00:01 bash
 12703 pts/1        01:13:00 zombie
 12704 pts/1        00:00:00 zombie <defunct>
 13143 pts/1        00:00:00 ps
proshb@proshb-vm:~/test$ kill 12703
proshb@proshb-vm:~/test$ ps
  PID TTY          TIME CMD
 2686 pts/1        00:00:01 bash
 13144 pts/1        00:00:00 ps
[1]+  Terminated                  ./zombie
proshb@proshb-vm:~/test$ r
```

`ps -> process`

`kill -> reap`

Behavior of Background Process

- `./hello &` (Hello world)

```
proshb@proshb-vm:~/test$ ./hello &
[1] 13199
proshb@proshb-vm:~/test$ Hello world
```

It does not matter which comes first

enter

```
proshb@proshb-vm:~/test$ ./hello &
[1] 13199
proshb@proshb-vm:~/test$ Hello world

[1]+  Done                  ./hello
proshb@proshb-vm:~/test$
```

Behavior of Background Process

- `./loop &` (Infinite loop)

```
proshb@proshb-vm:~/test$ ./loop &  
[1] 13231  
proshb@proshb-vm:~/test$  
proshb@proshb-vm:~/test$  
proshb@proshb-vm:~/test$  
proshb@proshb-vm:~/test$
```

Note

- If your code has some changes to requirement, please note it on the contents of e-mail.
- Submit : swe3019@csl.skku.edu
- Due : 4.16(Tue) 23:59:59