

Lab2

2016-18221 0|동현

1. 실행 결과

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test01
./sdriver.pl -t trace01.txt -s ./tsh -a "-p"
#
# trace01.txt - Properly terminate on EOF.
#
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$
```

사진 1. test01

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test02
./sdriver.pl -t trace02.txt -s ./tsh -a "-p"
#
# trace02.txt - Process builtin quit command.
#
```

사진 2. test02

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test03
./sdriver.pl -t trace03.txt -s ./tsh -a "-p"
#
# trace03.txt - Run a foreground job.
#
tsh> quit
```

사진 3. test03

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test04
./sdriver.pl -t trace04.txt -s ./tsh -a "-p"
#
# trace04.txt - Run a background job.
#
tsh> ./myspin 1 &
[1] (2399) ./myspin 1 &
```

사진 4. test04

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test05
./sdriver.pl -t trace05.txt -s ./tsh -a "-p"
#
# trace05.txt - Process jobs builtin command.
#
tsh> ./myspin 2 &
[1] (2405) ./myspin 2 &
tsh> ./myspin 3 &
[2] (2407) ./myspin 3 &
tsh> jobs
[1] (2405) Running ./myspin 2 &
[2] (2407) Running ./myspin 3 &
```

사진5. test05

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test06
./sdriver.pl -t trace06.txt -s ./tsh -a "-p"
#
# trace06.txt - Forward SIGINT to foreground job.
#
tsh> ./myspin 4
Job [1] (2415) terminated by signal 2
```

사진6. test06

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test07
./sdriver.pl -t trace07.txt -s ./tsh -a "-p"
#
# trace07.txt - Forward SIGINT only to foreground job.
#
tsh> ./myspin 4 &
[1] (2421) ./myspin 4 &
tsh> ./myspin 5
Job [2] (2423) terminated by signal 2
tsh> jobs
[1] (2421) Running ./myspin 4 &
```

사진7. test07

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test08
./sdriver.pl -t trace08.txt -s ./tsh -a "-p"
#
# trace08.txt - Forward SIGTSTP only to foreground job.
#
tsh> ./myspin 4 &
[1] (2430) ./myspin 4 &
tsh> ./myspin 5
Job [2] (2432) stopped by signal 20
tsh> jobs
[1] (2430) Running ./myspin 4 &
[2] (2432) Stopped ./myspin 5
```

사진8. test08

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test09
./sdriver.pl -t trace09.txt -s ./tsh -a "-p"
#
# trace09.txt - Process bg builtin command
#
tsh> ./myspin 4 &
[1] (2439) ./myspin 4 &
tsh> ./myspin 5
Job [2] (2441) stopped by signal 20
tsh> jobs
[1] (2439) Running ./myspin 4 &
[2] (2441) Stopped ./myspin 5
tsh> bg %2
[2] (2441) ./myspin 5
tsh> jobs
[1] (2439) Running ./myspin 4 &
[2] (2441) Running ./myspin 5
```

사진 9. test09

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test10
./sdriver.pl -t trace10.txt -s ./tsh -a "-p"
#
# trace10.txt - Process fg builtin command.
#
tsh> ./myspin 4 &
[1] (2452) ./myspin 4 &
tsh> fg %1
Job [1] (2452) stopped by signal 20
tsh> jobs
[1] (2452) Stopped ./myspin 4 &
tsh> fg %1
tsh> jobs
```

사진 10. test10

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test11
./sdriver.pl -t trace11.txt -s ./tsh -a "-p"
#
# trace11.txt - Forward SIGINT to every process in foreground process group
#
tsh> ./mysplit 4
Job [1] (2462) terminated by signal 2
tsh> /bin/ps a
    PID TTY      STAT   TIME COMMAND
      1178 tty2    Ssl+  0:00 /usr/lib/gdm3/gdm-x-session --run-script env GNOME
_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --systemd --session=ubuntu
      1180 tty2    Sl+   0:05 /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/use
r/1000/gdm/Xauthority -background none -noreset -keeptty -verbose 3
      1215 tty2    Sl+   0:00 /usr/libexec/gnome-session-binary --systemd --syst
emd --session=ubuntu
      1688 pts/0    Ss    0:00 bash
      1772 pts/0    S     0:00 ./tsh -p
      1773 pts/0    Z     0:00 [echo] <defunct>
      2457 pts/0    S+   0:00 make test11
      2458 pts/0    S+   0:00 /bin/sh -c ./sdriver.pl -t trace11.txt -s ./tsh -a
"-p"
      2459 pts/0    S+   0:00 /usr/bin/perl ./sdriver.pl -t trace11.txt -s ./tsh
-a -p
      2460 pts/0    S+   0:00 ./tsh -p
      2465 pts/0    R     0:00 /bin/ps a
```

사진 11. test 11

```

./sdriver.pl -t trace12.txt -s ./tsh -a "-p"
#
# trace12.txt - Forward SIGTSTP to every process in foreground process group
#
tsh> ./mysplit 4
Job [1] (2471) stopped by signal 20
tsh> jobs
[1] (2471) Stopped ./mysplit 4
tsh> /bin/ps a
    PID TTY      STAT   TIME COMMAND
    1178 tty2    Ssl+   0:00 /usr/lib/gdm3/gdm-x-session --run-script env GNOME
_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --systemd --session=ubuntu
    1180 tty2    Sl+    0:06 /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/use
r/1000/gdm/Xauthority -background none -noreset -keeptty -verbose 3
    1215 tty2    Sl+    0:00 /usr/libexec/gnome-session-binary --systemd --syst
emd --session=ubuntu
    1688 pts/0    Ss     0:00 bash
    1772 pts/0    S      0:00 ./tsh -p
    1773 pts/0    Z      0:00 [echo] <defunct>
    2466 pts/0    S+    0:00 make test12
    2467 pts/0    S+    0:00 /bin/sh -c ./sdriver.pl -t trace12.txt -s ./tsh -a
"-p"
    2468 pts/0    S+    0:00 /usr/bin/perl ./sdriver.pl -t trace12.txt -s ./tsh
-a -p
    2469 pts/0    S+    0:00 ./tsh -p
    2471 pts/0    T      0:00 ./mysplit 4
    2472 pts/0    T      0:00 ./mysplit 4
    2475 pts/0    R      0:00 /bin/ps a

```

사진 12. test12

```

donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test13
./sdriver.pl -t trace13.txt -s ./tsh -a "-p"
#
# trace13.txt - Restart every stopped process in process group
#
tsh> ./mysplit 4
Job [1] (2482) stopped by signal 20
tsh> jobs
[1] (2482) Stopped ./mysplit 4
tsh> /bin/ps a
    PID TTY      STAT   TIME COMMAND
    1178 tty2    Ssl+   0:00 /usr/lib/gdm3/gdm-x-session --run-script env GNOME
_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --systemd --session=ubuntu
    1180 tty2    Sl+    0:06 /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/use
r/1000/gdm/Xauthority -background none -noreset -keeptty -verbose 3
    1215 tty2    Sl+    0:00 /usr/libexec/gnome-session-binary --systemd --syst
emd --session=ubuntu
    1688 pts/0    Ss     0:00 bash
    1772 pts/0    S      0:00 ./tsh -p
    1773 pts/0    Z      0:00 [echo] <defunct>
    2477 pts/0    S+    0:00 make test13
    2478 pts/0    S+    0:00 /bin/sh -c ./sdriver.pl -t trace13.txt -s ./tsh -a
"-p"
    2479 pts/0    S+    0:00 /usr/bin/perl ./sdriver.pl -t trace13.txt -s ./tsh
-a -p

```

사진 13. test13(1)

```

2480 pts/0    S+    0:00 ./tsh -p
2482 pts/0    T    0:00 ./mysplit 4
2483 pts/0    T    0:00 ./mysplit 4
2486 pts/0    R    0:00 /bin/ps a
tsh> fg %1
tsh> /bin/ps a
  PID TTY      STAT   TIME COMMAND
  1178 tty2    Ssl+  0:00 /usr/lib/gdm3/gdm-x-session --run-script env GNOME
_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --systemd --session=ubuntu
  1180 tty2    Sl+   0:06 /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/use
r/1000/gdm/Xauthority -background none -noreset -keeptty -verbose 3
  1215 tty2    Sl+   0:00 /usr/libexec/gnome-session-binary --systemd --syst
emd --session=ubuntu
  1688 pts/0    Ss    0:00 bash
  1772 pts/0    S    0:00 ./tsh -p
  1773 pts/0    Z    0:00 [echo] <defunct>
  2477 pts/0    S+    0:00 make test13
  2478 pts/0    S+    0:00 /bin/sh -c ./sdriver.pl -t trace13.txt -s ./tsh -a
"-p"
  2479 pts/0    S+    0:00 /usr/bin/perl ./sdriver.pl -t trace13.txt -s ./tsh
-a -p
  2480 pts/0    S+    0:00 ./tsh -p
  2489 pts/0    R    0:00 /bin/ps a

```

사진 14. test13(2)

```

donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test14
./sdriver.pl -t trace14.txt -s ./tsh -a "-p"
#
# trace14.txt - Simple error handling
#
tsh> ./bogus
./bogus: Command not found
tsh> ./myspin 4 &
[1] (2498) ./myspin 4 &
tsh> fg
fg command requires PID or %jobid argument
tsh> bg
bg command requires PID or %jobid argument
tsh> fg a
fg: argument must be a PID or %jobid
tsh> bg a
bg: argument must be a PID or %jobid
tsh> fg 9999999
(9999999): No such process
tsh> bg 9999999
(9999999): No such process
tsh> fg %2
%2: No such job
tsh> fg %1

```

사진 15. test14(1)

```
tsh> fg %1
Job [1] (2498) stopped by signal 20
tsh> bg %2
%2: No such job
tsh> bg %1
[1] (2498) ./myspin 4 &
tsh> jobs
[1] (2498) Running ./myspin 4 &
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$
```

사진 16. test14(2)

```
donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test15
./sdriver.pl -t trace15.txt -s ./tsh -a "-p"
#
# trace15.txt - Putting it all together
#
tsh> ./bogus
./bogus: Command not found
tsh> ./myspin 10
Job [1] (2544) terminated by signal 2
tsh> ./myspin 3 &
[1] (2546) ./myspin 3 &
tsh> ./myspin 4 &
[2] (2548) ./myspin 4 &
tsh> jobs
[1] (2546) Running ./myspin 3 &
[2] (2548) Running ./myspin 4 &
tsh> fg %1
Job [1] (2546) stopped by signal 20
tsh> jobs
[1] (2546) Stopped ./myspin 3 &
[2] (2548) Running ./myspin 4 &
tsh> bg %3
%3: No such job
tsh> bg %1
[1] (2546) ./myspin 3 &
tsh> jobs
[1] (2546) Running ./myspin 3 &
[2] (2548) Running ./myspin 4 &
tsh> fg %1
tsh> quit
```

사진 17. test15

```

donghyun@donghyun-VirtualBox:~/dev/2022_sp_linker/shlab$ make test16
./sdriver.pl -t trace16.txt -s ./tsh -a "-p"
#
# trace16.txt - Tests whether the shell can handle SIGTSTP and SIGINT
#      signals that come from other processes instead of the terminal.
#
tsh> ./mystop 2
Job [1] (2562) stopped by signal 20
tsh> jobs
[1] (2562) Stopped ./mystop 2
tsh> ./myint 2
Job [2] (2565) terminated by signal 2

```

사진18. test16

2. 구현 방법

- eval

`eval`은 `input command`를 해석하고 프로세스 전반의 처리를 담당한다. 교재의 함수를 응용하여 구현하였으며, 입력받은 문자열을 파싱한 뒤 성공했다면 `builtin_cmd` 가 아닌 경우 알맞은 처리를 한다. `execve`를 통해 실행하고 난 뒤 `addjob`을 이용해 `jobs` 리스트에 추가해준다.

`execve` 함수가 실패하면 `child` 프로세스가 종료되었는지 확인해야 했는데 `SIGCHLD`가 자식임을 명확하게 하기 위해서 `SIGCHLD`를 `blocking`하고 `fork`를 수행한 후 `execve`를 해주었다. 또한 `sigint_handler` 등에서 `pid`가 필요한데, 이를 위해 `setpgid`를 통해 프로세스의 `pid`를 변경해주었다. 마지막으로 `background`와 `foreground`에서 프로세스 실행 제한이 다르기 때문에 `background`일 경우 `waitfg`를 호출해주었다.

- builtin_cmd

`Quit`, `jobs`, `fg`, `bg` 처리를 담당하는데, 정상 실행인 경우 1을, 아닌 경우 0을 리턴한다. `quit`이면 종료, `jobs`에서는 `listjobs`, `bg` 또는 `fg`에서는 `do_bgfg`를 호출해서 각 `running` 상태로 바꿔주었다.

- do_bgfg

`builtin cmd`인 `bg`, `fg`를 실행하는데 인자로 `jid` 혹은 `pid`를 추출하여 `job`을 받아온다. 이 `job`의 `pid`를 이용해서 커멘드에 맞게 `state`를 변경해주었다. 인자가 없거나 `job`이 없는 경우, `pid`로 `job`을 받아올 수 없는 경우 등을 에러처리 해주었다.

- waitfg

`foreground` 프로세스가 끝날 때까지 `while`을 이용해서 무한루프를 수행한다. `pid != fgpid(jobs)`의 조건이 참이되기 전까지 `sleep`을 통해 주기적으로 확인한다.

- sigchld_handler

`SIGCHLD` 시그널을 처리하는 함수로, `child process`가 종료된 상황, 정지된 상황을 처리한다. 정지 상황이면 `status`를 변경해주고 종료 상황이면 `deletejob`을 통해 지워준다.

- sigint_handler

`SIGINT` 시그널을 처리한다. 현재 `foreground`의 `pid`를 가져온 후 `kill`을 통해 `SIGINT`를 발생시킨다.

- `sigstp_handler`

SIGSTOP 시그널을 처리한다. 현재 foreground의 pid를 가져온 후 kill을 통해 SIGSTP를 발생시킨다.

3. 어려웠던 점

`eval`을 구현하면서 시그널을 `block`하면서 실행 순서를 맞추어주는 작업이 어려웠다. 순서를 맞추지 못할 경우 실행에 실패한 프로세스가 `jobs`목록에 추가되는 경우가 있어서 `block`과 `unblock`을 적절히 사용해주어야 했다. 또한 함수명들이 익숙하지 않아 여러번 실수를 반복하며 찾아보아야 했고 결과물을 보면서 로그 출력을 맞추어주는 과정 또한 어려웠다.

4. 새롭게 배운 점

`shell`의 내부 `signal handling`이나 `fork`후 `process`관리 등을 직접 경험해보면서 동작을 좀 더 깊게 이해할 수 있었다. 실패한 프로세스들이 남아있지 않도록 리소스를 관리하는 방식을 경험하면서 앞으로 `shell script` 등을 작성할 때에도 구조적인 부분을 고민할 수 있게 될 것이라고 생각한다.