

2018 시스템 프로그래밍

- lab06 -

제출일자	2018.11.19
분 반	01
이 름	함지희
학 번	201702087

Trace 05 - 06

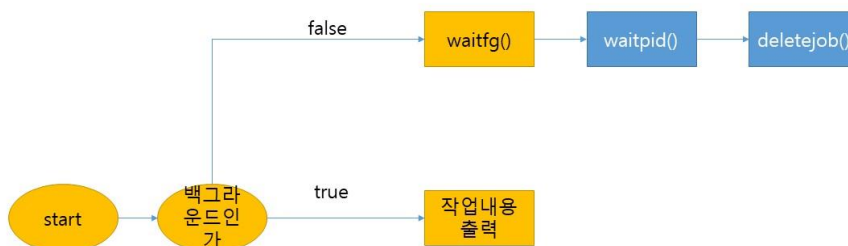
```
b20170208/02018-sp:~/shlab-handout$ ./sdriver -V -t 06 -s ./tshref
Running trace06.txt...
Success: The test and reference outputs for trace06.txt matched!
Test output:
#
# trace06.txt - Run a foreground job and a background job.
#
tsh> ./myspin1 &
(1) (21563) ./myspin1 &
tsh> ./myspin2 1

Reference output:
#
# trace06.txt - Run a foreground job and a background job.
#
tsh> ./myspin1 &
(1) (21572) ./myspin1 &
tsh> ./myspin2 1
```

```
b20170208/02018-sp:~/shlab-handout$ ./sdriver -V -t 06 -s ./tsh
Running trace06.txt...
Success: The test and reference outputs for trace06.txt matched!
Test output:
#
# trace06.txt - Run a foreground job and a background job.
#
tsh> ./myspin1 &
(1) (21387) ./myspin1 &
tsh> ./myspin2 1

Reference output:
#
# trace06.txt - Run a foreground job and a background job.
#
tsh> ./myspin1 &
(1) (21396) ./myspin1 &
tsh> ./myspin2 1
```

각 trace 별 프로우 차트



Trace 해결 방법 설명

```
189     if(!bg){
190         waitfg(pid, 0);
191     }
192     else{
193         printf("(%d) (%d) %s",pid2jid(pid),pid,cmdline);
194     }
195 }
```

만약 bg가 0이면 자식프로세스가 실행 중이라는 의미이므로 waitfg함수를 사용하여 자식프로세스가 끝날 때까지 기다린다.

```
215 void waitfg(pid_t pid, int output_fd)
216 {
217     pid = waitpid(pid, NULL, output_fd);
218     deletejob(jobs , pid);
219     return;
220 }
```

waitfg 함수의 매개변수로는 자식 프로세스의 pid와 0이 들어가는데 이걸 waitpid함수에 넣는다. waitpid의 세번째 매개변수에 0이 들어가야하는 이유는 옵션에 0이 들어가면 wait와 같은 역할 즉, 자식 프로세스가 종료될때까지 block하기 때문이다. 두번째 매개변수에 NULL이 들어가는 이유는 원래 거기에 상태정보를 저장하는 변수가 있어야하지만 현재 코드엔 상태정보를 저장하는 변수가 없어서 일단 NULL로 하였다. NULL이 아니었다면 이 변수를 통해 프로세스의 상태정보를 알아낼 수 있을 것이다. waitpid의 반환값을 종료된 자식프로세스의 ID이다.

deletejob함수를 통해 자식프로세스를 작업리스트에서 삭제한다.

자식프로세스를 처리한 뒤에는 백그라운드의 작업을 수행하고 그 작업내용을 출력한다. 출력은 작업id, 프로세스 id, 커맨드가 출력되게끔 한다.

Trace 07

```
b20170208702018-sp:~/shlab-handout$ ./sdriver -V -t 07 -s ./tshref
Running trace07.txt...
Success: The test and reference outputs for trace07.txt matched!
Test output:
#
# trace07.txt - Use the jobs builtin command.
#
tsh> ./myspin1 10 &
(1) (16146) ./myspin1 10 &
tsh> ./myspin2 10 &
(2) (16148) ./myspin2 10 &
tsh> jobs
(1) (16146) Running ./myspin1 10 &
(2) (16148) Running ./myspin2 10 &

Reference output:
#
# trace07.txt - Use the jobs builtin command.
#
tsh> ./myspin1 10 &
(1) (16156) ./myspin1 10 &
tsh> ./myspin2 10 &
(2) (16158) ./myspin2 10 &
tsh> jobs
(1) (16156) Running ./myspin1 10 &
(2) (16158) Running ./myspin2 10 &
```

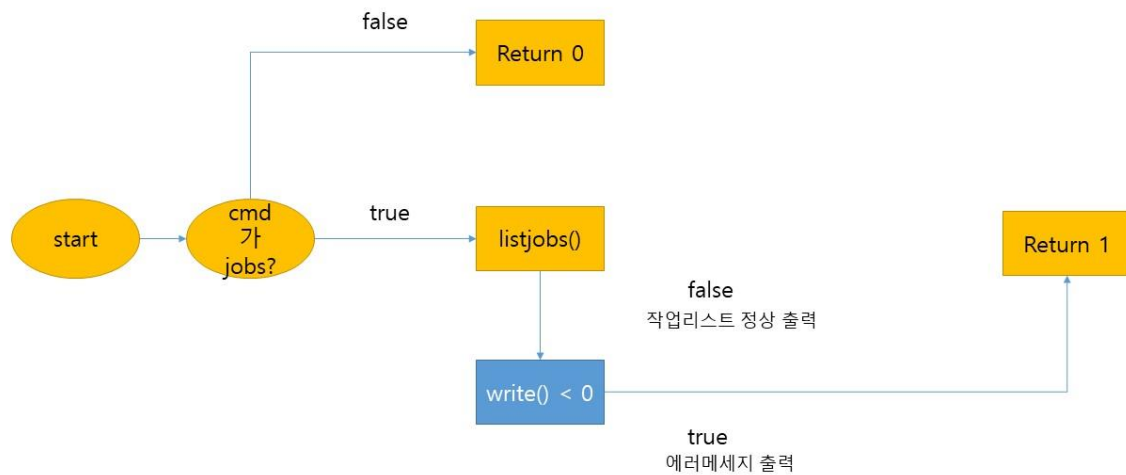
```

b201702087@2018-sp:~/shlab-handout$ ./sdriver -V -t 07 -s ./tsh
Running trace07.txt...
Success: The test and reference outputs for trace07.txt matched!
Test output:
#
# trace07.txt - Use the jobs builtin command.
#
tsh> ./myspin1 10 &
(1) (16847) ./myspin1 10 &
tsh> ./myspin2 10 &
(2) (16849) ./myspin2 10 &
tsh> jobs
(1) (16847) Running    ./myspin1 10 &
(2) (16849) Running    ./myspin2 10 &

Reference output:
#
# trace07.txt - Use the jobs builtin command.
#
tsh> ./myspin1 10 &
(1) (16858) ./myspin1 10 &
tsh> ./myspin2 10 &
(2) (16860) ./myspin2 10 &
tsh> jobs
(1) (16858) Running    ./myspin1 10 &
(2) (16860) Running    ./myspin2 10 &

```

각 trace 별 프로우 차트



Trace 해결 방법 설명

```
207     else if(!strcmp(cmd, "jobs")){
208         listjobs(jobs,0);
209         return 1;
210     }
```

jobs를 입력받았을때 실행하도록 한다. cmd와 "jobs"가 같으면 0이 나올것이고 이것은 즉 같다는 뜻이기 때문에 조건문 안으로 들어간다. listjobs함수를 이용하여 작업리스트를 출력하고 builtin_cmd이기 때문에 1을 반환한다.

```
456     if (jobs[i].pid != 0) {
457         sprintf(buf, "(%d) (%d) ", jobs[i].jid, jobs[i].pid);
458         if(write(output_fd, buf, strlen(buf)) < 0) {
459             fprintf(stderr, "Error writing to output file\n");
460             exit(1);
461         }
```

```
switch (jobs[i].state) {
    case BG:
        sprintf(buf, "Running  ");
        break;
    case FG:
        sprintf(buf, "Foreground ");
        break;
    case ST:
        sprintf(buf, "Stopped  ");
        break;
    default:
        sprintf(buf, "listjobs: Internal error: job[%d].state=%d ",
            i, jobs[i].state);
}
```

```
482     sprintf(buf, "%s", jobs[i].cmdline);
```

listjob함수에서는 pid가 0이 아니면 그 프로세스의 ID와 작업ID를 출력한다. switch문으로 현재 프로세스의 상태를 출력하고 그 다음엔 커맨드도 출력한다. write함수를 이용해 잘 쓰기가 되었는지 확인한다.