**유닉스 프로그래밍 #2 과제**

***컴퓨터공학과***

***12171619***

***박윤수***

1.

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

우선 child\_exist는 자식이 존재하는지에 대한 int 변수로 0일 때 자식 프로세스가 존재하지 않고, 1일 때 자식 프로세스가 존재한다.

sh\_int 함수는 SIGINT와 SIGQUIT 시그널 처리 함수인데, 자식 프로세스가 존재하지 않으면 프롬프트까지 출력하고, 존재하면 개행문자만 출력한다.

이후 메인함수에서 sigaction을 만드는데 flag를 SA\_RESTART로 선언하여 만약 시스템콜 실행 중 SIGINT나 SIGQUIT가 발생하면 이를 처리한 후 다시 시스템콜을 실행하도록 한다.

myshell은 자식 프로세스가 항상 종료된 후 와일문을 진행하기 때문에 와일문의 맨 처음 부분에 child\_exist를 0으로 초기화한다.

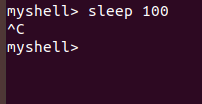
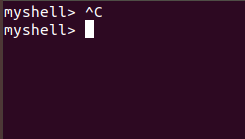
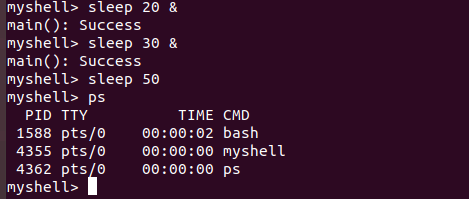
텍스트이(가) 표시된 사진

자동 생성된 설명

이후 fork부분에서 자식 프로세스의 경우 SIGINT와 SIGQUIT의 처리를 원래의 방식으로 되돌린다. 부모 프로세스, 즉 myshell의 경우 자식이 존재하므로 child\_exist를 1로 하고 자식 프로세스가 종료될 때까지 기다린다. 만약 기다리는 도중 시그널이 발생하면 위의 처리 함수에서 개행문자만 출력한다.

(만약 sleep 10 이후 SIGINT를 발생시키면 부모와 자식 프로세스 둘 다 SIGINT를 처리하는지 둘중 하나만 처리하는지에 대해선 차후 고민해 봐야 할 것 같다. 이번 프로젝트에서는 우선 둘 다 SIGINT를 처리하는 것으로 생각하였다.)

고아 프로세스 생성 원리를 이용하여 백그라운드를 구현했기 때문에 좀비 프로세스가 발생하지 않는다. (물론 손주 프로세스가 종료된 이후 init 프로세스가 입양하여 wait하기 전까진 좀비 프로세스이지만 매우 빠르게 처리되기 때문에 우리가 확인할 수 없다.) 따라서, SIGCHLD를 이용하여 좀비 프로세스가 생성되지 않도록 만드는 작업은 생략하였다.



1, 2, 3번 테스트를 진행한 결과이다.