Operating Systems

Project #1

(Draft report)

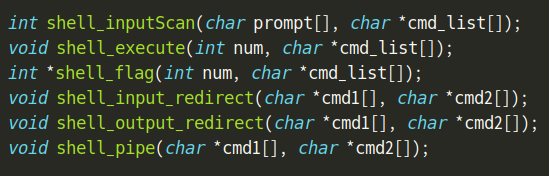
미완성 보고서

2018045214 | 최 준 희

2021.03.28

Simple Shell algorithm

함수



프로그램 흐름

1. 프로그램이 실행되면, 바로 자식프로세스를 생성하여 자식프로세스에서 입력을 받는다.

2. 입력 받은 내용을 inputScan을 통해 토큰화

3. 토큰화된 단어 포인터 배열을 가지고 execute를 실행한다.

4. execute 에서는 shell\_flag 함수를 사용하여 리다이렉션이나 파이프가 있는지 확인한다.

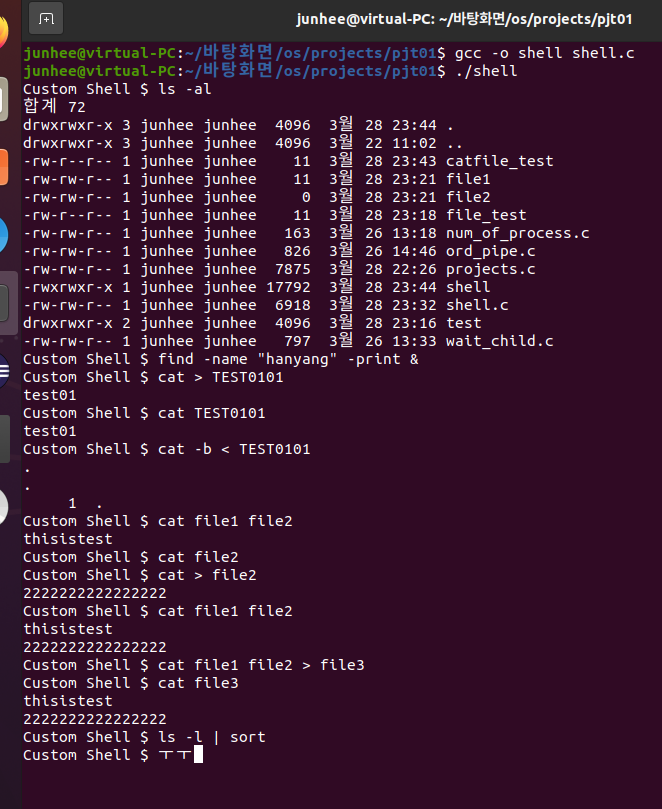
5-1. 리다이렉션이 있는 경우에는 input\_redirect, output\_redirect 함수를 호출한다.

5-2. input\_redirect에서는 파일 디스크립터를 이용해 새로운 파일을 open하고, dup2를 사용하여 STDOUT\_FILENO (표준 출력)을 파일 디스크립터가 가르키도록 하고, execvp를 통해 명령어를 실행한다.

5-3. output\_redirect에서도 파일 디스크립터를 통해open을 사용하며, 내용을 읽어올 파일을 read하여 저장, 2번에서 사용했던 inputScan을 통해 내용을 토큰화 하고, 그 내용을 가지고 execvp를 통해 실행할 수 있도록 한다.

6. 파이프가 있는 경우엔 두번의 fork를 통해 표준 출력을 받아서 담아두었다가, 표준 출력을 표준 입력에 집어넣어서 연산이 이루어 지도록 기획했다.

실행 화면



* Cat -b < test0101 과 같이 리다이렉션이 아직 버그가 있습니다…
* 파이프가 작동하지 않습니다.
* 4월 1일까지 디버깅 하겠습니다.

소스 코드