

Simple shell 프로그램 (myshell)

본 프로젝트에서는 자신의 셸(myshell)을 설계한다. 이 셸의 사용법은 다음의 두 가지이다.

(가) myshell 로 시작하여 ctrl-D까지의 명령어를 처리한다.

(예) myshell 의 프롬프트는 \$ 라고 가장하자.

myshell

\$ echo test program

\$ ls

ctrl-D

(나) myshell 의 인수로 임의의 명령어를 사용하여 수행시킨다.(-c 옵션)

(예) myshell -c string

myshell -c "echo test program ; ls -F"

다음은 구현하고 각각을 테스트시오.

(1) 위의 두 방식으로 동작하는 셸을 만드시오.

(2) 다음과 같은 background job이 동작하도록 하시오.

\$ grep the file &

\$

(3) I/O redirection을 구현하시오.(multiple redirection 가능)

표준입력, 표준출력, 표준에러를 이용하는 간단한 프로그램 prog를 작성하여 테스트하시오.

\$ prog < inputfile

\$ prog > outputfile

\$ prog 2> errorfile

\$ prog < inputfile > outputfile 2> errorfile

(4) 파이프의 기능을 구현하시오.

\$ ls | sort

(5) 다단계 파이프 기능을 구현하시오. (다단계 제한 없음)

\$ cat /etc/passwd | sort | grep kim

참고 : 모든 명령어 라인의 토큰들은 “ ” 로 떨어져 있다고 가정해도 된다.

fork(), exec계열, pipe() 시스템콜을 사용하고 system(), popen() 등은 사용하지 마시오.

<testcase example>

./myshell -c "echo test program"

./myshell -c "ls -l -F"

/bin/ps

/bin/ls -l -F

/bin/echo hello

/bin/hostname

/bin/true

grep ZOK <ifile

ls -l >ofile

echo catOfile

cat ofile

ls -ERR 2> efile

echo catEfile

cat efile

grep ZOK <ifile >ofile 2> efile

echo catOfile

cat ofile

echo catEfile

cat efile

cat /etc/passwd | sort | grep ksi | grep ksita