

운영체제

-mysh2 구현-

소프트웨어학과

201520860

노근탁

Parser : .unit_test의 기준인 "W"과 " "가 나오면 명령어를 잘라야 하므로 두가지 경우를 나누어서 strtok 함수를 이용하여 구현하였고 token을 저장하기 위해서 argv 배열을 할당해야하는데 나중에 free를 위해서 토큰의 개수에 맞게 할당을 해야해서 argc를 구하는 함수도 같이구현하였습니다. Unit_test와 test 5 까지는 정상적으로 작동하였으나 테스트7에서 올바르게 작동하지 않았습니다

dose_exfile_exists : stat 함수를 이용하여 파일의 정보를 받아서 S_IXUSR를 이용하여 &연산으로 실행가는 파일인지를 확인하였습니다.

Fetch_command : argv[0]을 받아 pwd인지 cd 인지를 확인하여 맞으면 해당 배열을 리턴하게하였습니다.

Do_pwd : getcwd를 따라 할당한 buf에 저장하여 working directory를 출력하게 하였습니다.

Do_cd : chdir함수를 활용하여 cd를 구현하였고 chdir 리턴 값이 -1일 경우 제대로 수행이 안된 경우이므로 file 이나 directory가 존재하지 않는 경우, path가 file 이나 directory가 아닌 경우를 나누어서 각각 할당된 리턴값을 반환하게 했습니다.

Err_cd : int 형을 전달받아 해당 에러에 맞는 오류 메시지를 fprintf를 활용하여 stderr로 출력하게 하였습니다

.main의 to do 부분 : argv[0]가 실행가능하다면 fork로 자식 프로세스를 만들고 execvp 를 사용하여 argv[0]를 실행할 수 있도록 하였습니다.

고찰 : 처음에는 parser를 구현하는데 애를 많이 먹었습니다. 구현을 한뒤에도 메모리 관리가 되지 않아서 많은 시간을 할당해야 했습니다. 이후에 테스트로 넘어간 부분은 구글링을 통해 fork와 exec계열 함수의 사용법을 익히고 직접 활용할 수 있게 되어 조금더 개념적인 부분 보다 적용가능한 부분을 배울수 있어서 좋았습니다. 그러나 parser가 unit_test는 통과하나 test7에서 parser 기능을 제대로 수행하지 못한점은 더욱 생각을 해봐야할 것 같습니다. 제가 생각해보았을 때 논리적으로는 맞는데 실행이 안되는 부분이라 제 생각이 잘못된 것일 수도 있지만, unit_test는 통과하는데 test에서 기능을 수행하지 못하는 이유는 찾지 못하였습니다. 개인적으로 이번과제에서 아쉬웠던 점은 unit_test를 parser가 통과를 해도 test에가서 줄바꿈 때문에 작동을 제대로 하지못하여 다시 손을 조금 봐야하는 점이 혼란스러웠던 것 같습니다. 또한 포인터와 메모리 관리 측면에서 제가 더 공부를 해야겠다는 생각이 들었습니다.