상명대학교 컴퓨터과학부 "EA0014: 유닉스프로그래밍" 실습 지침서

- 실습 번호 : lab-05

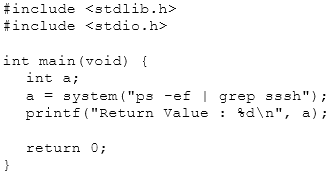
- 실습 디렉토리 : ~/unix/lab-05 이름 : 임현

- 실습 날짜 : 2017년 4월 21일 분반 : 1분반

- 실습 제목 : Process

- 실습 내용 : 학번 : 201511054

1. (5-1.c, 5-1) system()을 이용하는 아래의 프로그램 5-1.c를 작성하고 5-1로 컴파일하여 수행한 결과를 적으시오.

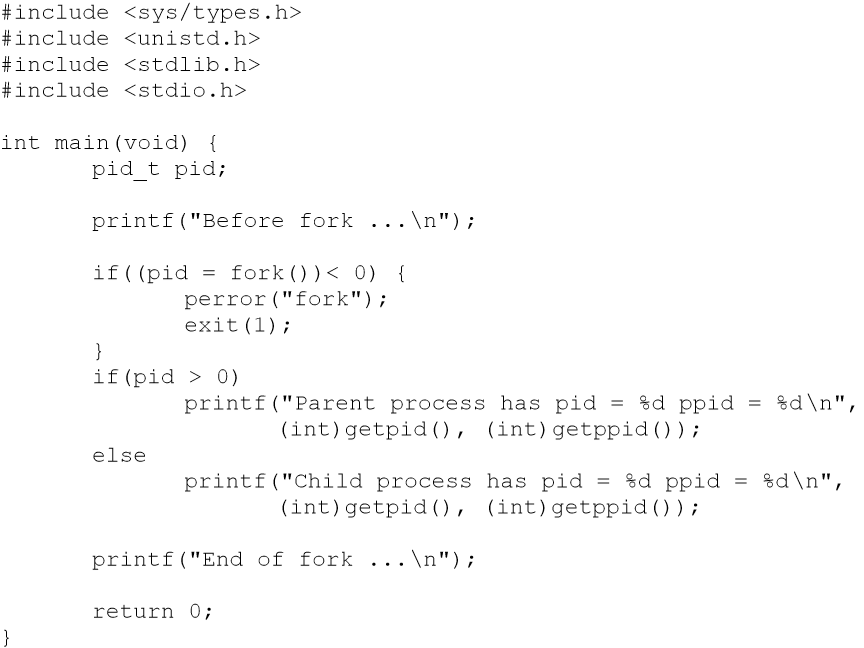


수행결과는 아래와 같고,

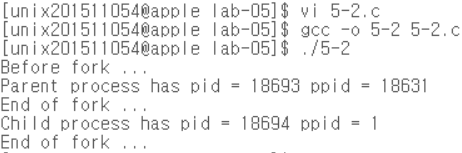
기존 명령이나 실행 파일명을 인자로 받아 쉘에 전달하는 system 함수 입니다.



2. (5-2.c, 5-2) fork()를 이용하여 새로운 프로세스를 만드는 아래 프로그램을 5-2.c로 작성하고 이를 컴파일하여 5-2를 만들고 수행하여 결과를 보이시오.



수행 결과는 아래와 같습니다.



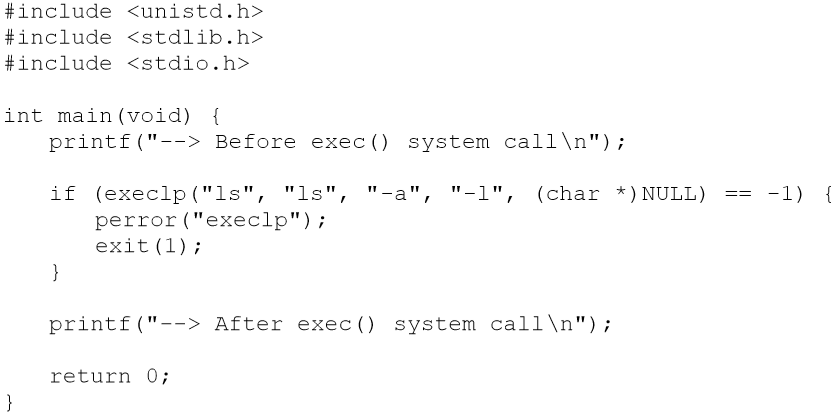
fork 함수를 호출하며 새로운 프로세스(자식 프로세스)를 생성하는 함수입니다.

생성한 자식 프로세스의 메모리 공간은 부모 프로세스의 메모리 공간을 그대로 복사해 만들고,

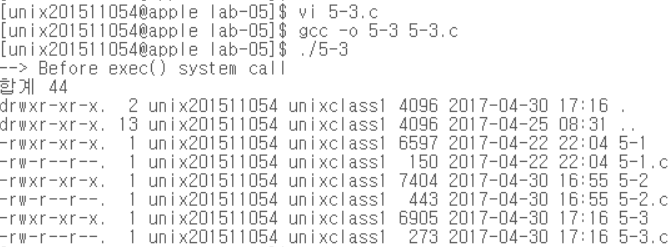
fork함수는 부모 프로세스에는 자식 프로세스의 PID를 리턴하고,

자식 프로세스에는 0을 리턴합니다.

3. (5-3.c, 5-3) 아래와 같이 exec()를 테스트하는 프로그램 5-3.c를 작성하고 이를 컴파일 하여 5-3을 만들고 실행하여 결과를 확인하고 그에 대해 설명하시오.



실행 결과는 아래와 같습니다.

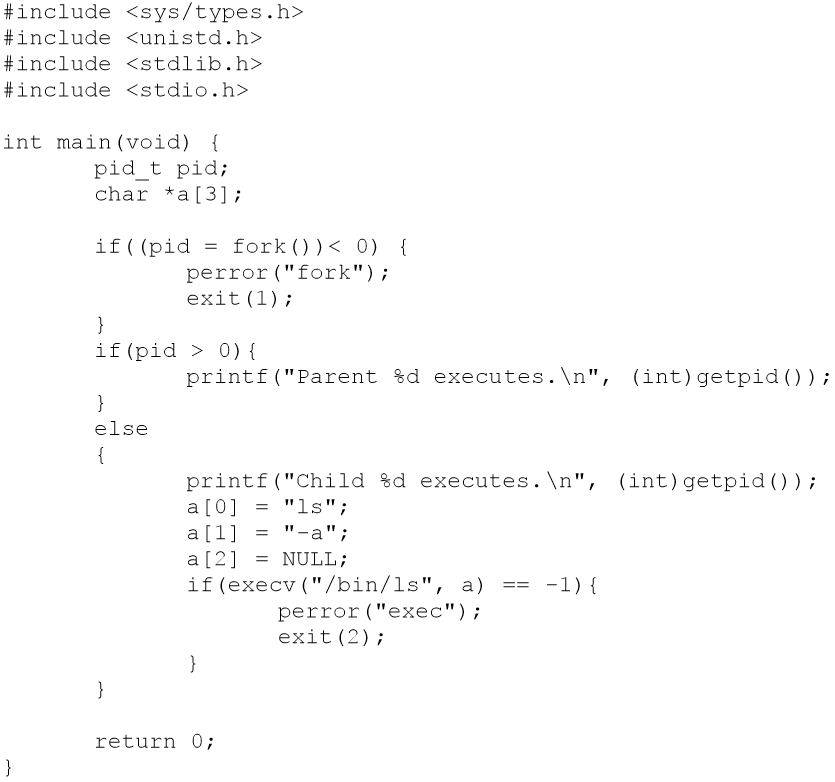


exec함수군은 사용하게 될 경우 명령이나 실행 파일을 실행할 수 있습니다.

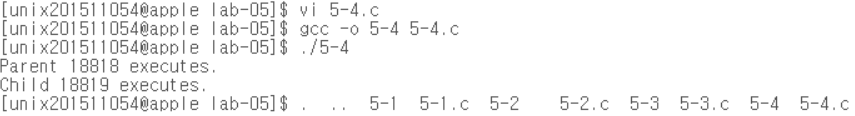
exec 함수군은 인자로 받은 다른 프로그램을 자신을 호출한 프로세스의 메모리에 덮어씁니다.

따라서 위와 같이 덮어쓴 프로그램(“ls 명령”)이 실행된 것 입니다.

4. (5-4.c, 5-4) 아래와 같이 fork()와 execv()를 테스트하는 프로그램 5-4.c를 작성하고 이를 컴파일하여 5-4를 만들고 실행하여 결과를 확인하고 그에 대해 설명하시오.



실행 결과는 아래와 같습니다.



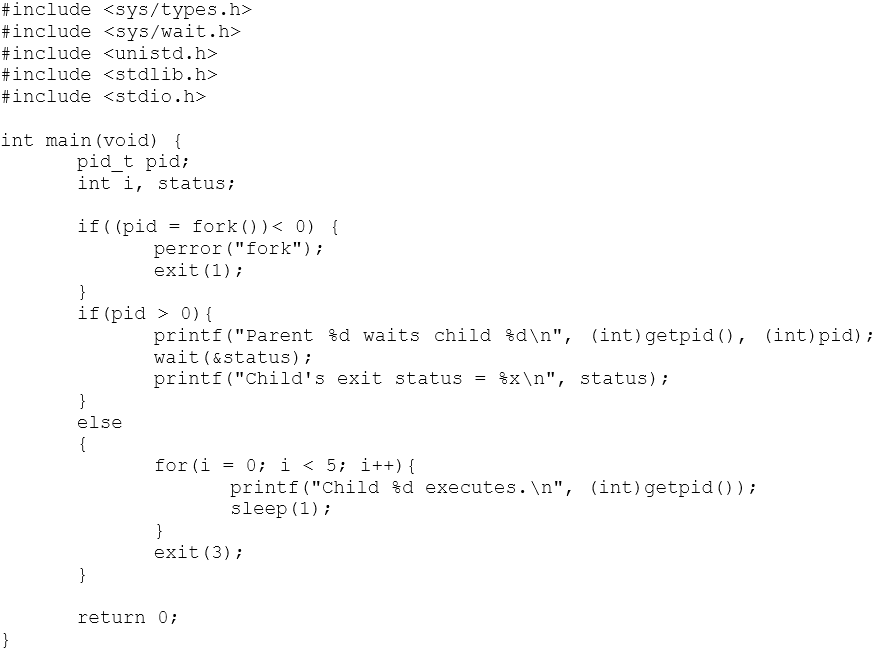
위 실행 결과 또한 3번처럼 exec군 함수를 사용한 것 입니다.

하지만 ‘execlp’ 함수를 사용했다는 점과, ‘execv’ 함수를 사용했다는 점이 다릅니다.

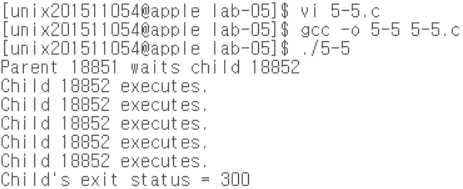
그리고 다른 점은 부모 프로세스에서 자식 프로세스를 생성해서,

자식 프로세스에서 execv 함수를 사용했다는 점 입니다.

5. (5-5.c, 5-5) 아래와 같이 wait()를 테스트하는 프로그램 5-5.c를 작성하고 이를 컴파일 하여 5-5을 만들고 실행하여 결과를 확인하고 그에 대해 설명하시오.



실행 결과는 아래와 같습니다.

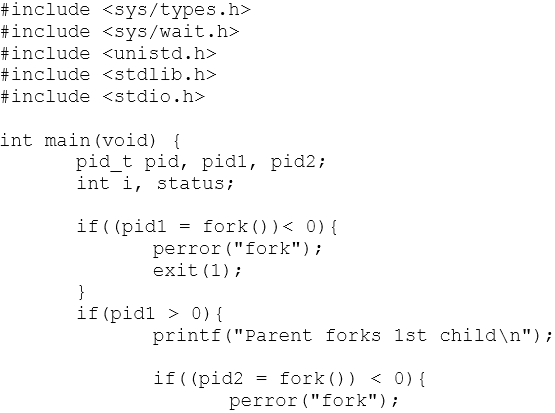


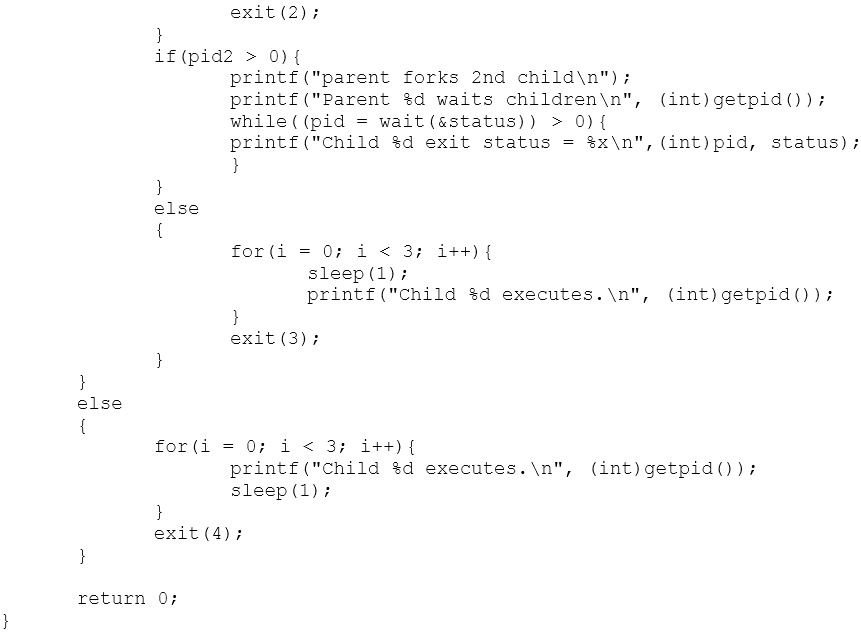
위 프로그램은 부모 프로세스가 자식 프로세스의 종료를 기다려 주기 위해 wait를 쓴 모습입니다.

wait를 쓰지 않을 경우 좀비 프로세스가 발생할 수 있습니다.

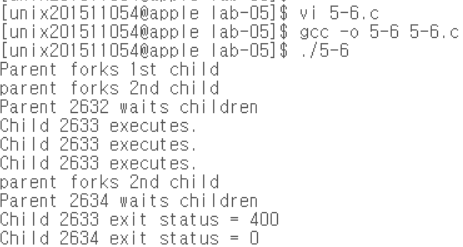
좀비 프로세스란 부모 프로세스가 자식 프로세스보다 먼저 종료된 경우로, 자식 프로세스가 자원을 반납할 수 없는 상태가 되므로, 일반적인 방법으로 제거할 수 없게 됩니다.

6. (5-6.c, 5-6) 아래의 프로그램을 5-6.c로 작성하고 이를 컴파일하여 5-6을 만들고 실행 하여 결과를 보이고 결과에 대해 설명하시오.





실행 결과는 아래와 같습니다.



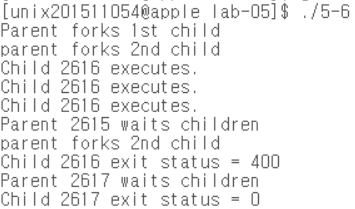
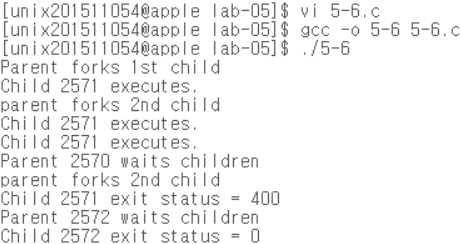
부모 프로세스가 1번째 자식 프로세스를 호출하고, 2번째 자식 프로세스를 호출한 상태입니다.

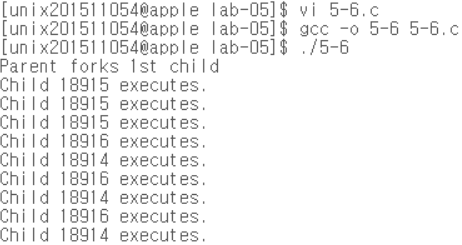
부모 프로세스가 1번째 자식 프로세스를 기다리다가,

종료가 되자 2번째 자식 프로세스를 실행 시키고 2번째 자식 프로세스 또한 기다리다가,

자식들의 종료 상태를 표시하는 모습입니다.

그리고 6번의 실행결과는 실행할 때 마다 달라지는데, 그 이유는 잘 모르겠습니다. 죄송합니다.





끝.