프로세스 생성 실습보고서

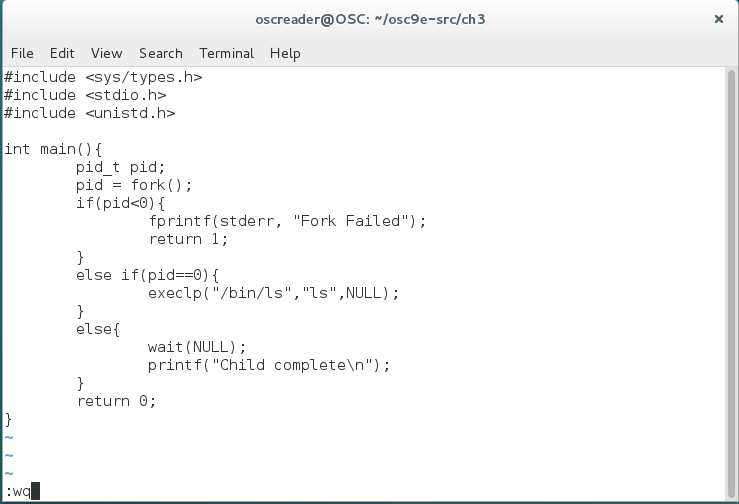
소프트웨어융합학과 2016111566 이나경

<파일 생성>



vi 편집기로 test.c 코드를 작성한다.

<코드 설명>



fork() 함수는 현재 실행중인 프로세스를 복사하여 실행한다.

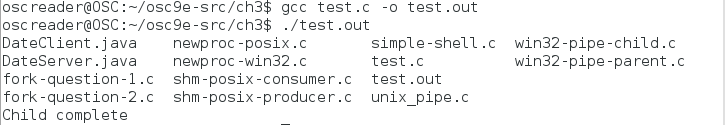
프로세스ID는 pid 변수에 저장되고

pid가 음수일 경우엔 fork() 에러 메시지 출력

pid가 0일 경우 child process로 ls 실행

그 외의 경우 parent process이므로 “child complete” 메시지 출력

<실행 결과>



컴파일 후 실행한 결과

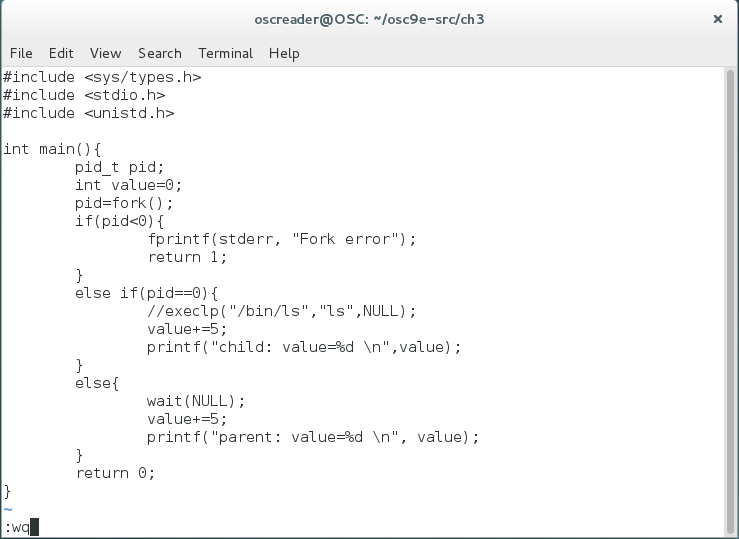
fork() 후 생성된 child가 먼저 실행되고 ( ls 실행 )

parent는 child의 종료를 기다린다. ( wait() 후 child complete )

즉, 아래 순서로 진행

[child 실행 – ls 실행 - child 종료 – parent 실행 – child complete 출력 – parent 종료]

<test2.c 실습>



코드를 수정하여 실행해본다.



자식, 부모 프로세스에서 모두 value가 5가 되었다.

이는 자식, 부모 프로세스가 동시에 실행된다는 것을 알려준다.

<정리>

1. 프로세스는 트리 구조로 되어 있다. 즉, 부모 프로세스가 자식 프로세스를 만든다.
2. 자식 프로세스는 부모로부터 권한과 스케줄링 속성을 상속하며 열린 파일과 같은 특정 리소스도 상속한다.
3. 시스템 콜의 fork() 함수를 호출하면 부모 프로세스는 자신과 똑같은 자식 프로세스를 생성한다. 운영체제가 정해준 고유 번호인 pid값으로 프로세스를 식별할 수 있는데, fork() 함수는 부모 프로세스에겐 자식 프로세스의 pid를, 자식프로세스에겐 0을 반환한다.
4. 부모 프로세스와 자식 프로세스는 동시에 작동한다.