실습 과제5

사이버보안학과

201620641

유 상 정

1. 과제 소개

* 조건을 만족하는 thread 동기화 프로그램 작성

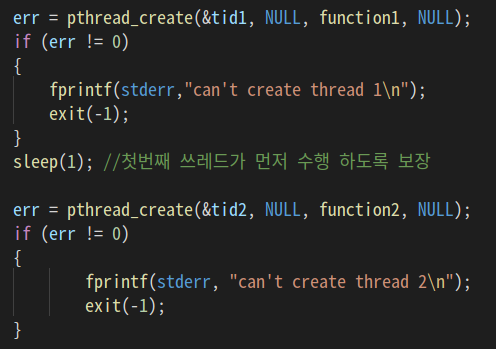
조건

* 2개의 thread 사용
* 각 thread는 count값을 0부터 증가시키고 11까지 그 값을 출력해주는 function1, function2를 실행
* count <=3, 8 <= count <= 10 일 때, 쓰레드1번이 출력, 나머지 쓰레드2번이 출력
* 조건변수사용

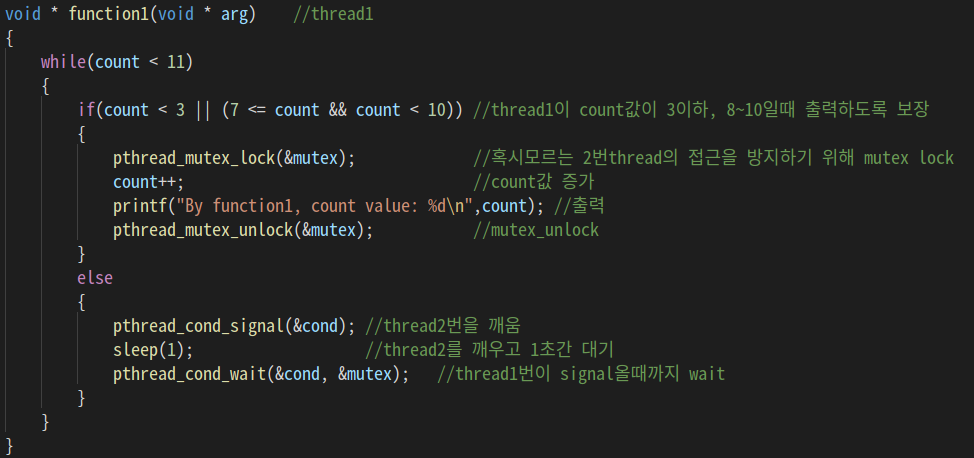
1. 스레드의 동기화가 필요한 이유

* 스레드들이 동일한 data에 접근을 할 때, 동기화가 없다면 race condition 상황이 발생하게 된다. 여기서 race condition이란, 두 개 이상의 concurrent한 스레드들이 동일한 자원을 놓고 서로 사용하려고 경쟁하는 상황을 말한다. 이 경우에, 스레드의 실행 순서를 잘 조절해주지 않으면 비정상적인 상태가 나타날 수 있다. 이러한 race condition상황을 막기위해서 스레드들 간의 동일한 data의 접근 순서를 조절해 주어야한다. 즉, 동기화가 필요하다.

1. 코드 설명



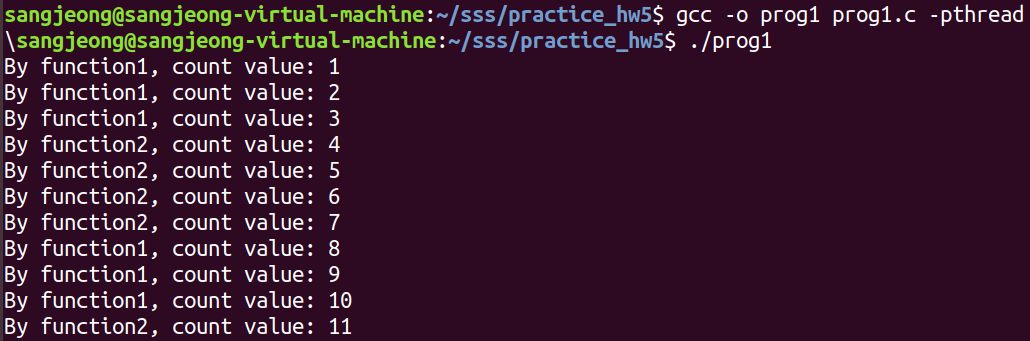
pthread\_create 함수를 통해서 두 개의 thread1, 2를 생성하고 각 thread가 function1, 2를 수행하게 한다. 이 때, 공유자원인 count에 thread1먼저 접근할 것을 보장하기위해서 sleep(1)을 이용하여 thread1 생성 후 1초간 대기한다. 공유자원인 count는 전역변수로서 0으로 초기화 되어있다.



Function1을 살펴보면 전체적으로 count < 11 인경우에 while문을 돌고 count < 3 또는 7<= count < 10 인경우에 mutex\_lock을 사용하여 count의 독점권을 얻고 count 증가 후 count값을 출력한다. 그리고 mutex\_unlock을 통해서 독점권을 해제한다. 그리고 나머지의 경우에는 cond\_signal을 통해서 thread2를 깨우고 1초간 대기 후, 자신은 thread2가 signal을 보낼 때까지 wait한다.



Function2의 경우도 function1과 유사하지만, count값이 3<= count <7 또는 count ==10인 경우에 mutex\_lock을 통해서 독점권을 얻고, count 증가, 출력 후 mutex\_unlock을 통해서 독점권을 해제한다. 그리고 최종 경우인 count 11을 출력한 후에는 thread2가 thread1을 깨우고, 함수를 빠져나간다. 최종적으로 pthread\_join함수를 통해서 각 thread의 종료를 기다린 후, 각 thread가 종료되면 pthread\_cond\_destroy 함수와 pthread\_mutex\_destroy 함수를 통해서 공유변수와 mutex를 파괴한다.



실행결과를 보면 조건 변수와 mutex를 사용하여 공유 자원 count의 접근이 문제없이 이루어졌다. 즉, 쓰레드 동기화가 잘 이루어 졌음을 알 수 있다.