**Chapter 19. Windows에서의 쓰레드 사용**

1. 다음 중 커널 오브젝트에 대한 설명으로 옳지 않은 것을 모두 고르면?
   1. 커널 오브젝트는 운영체제가 생성하는 리소스들의 정보를 저장해 놓은 데이터 블록이다.
   2. 커널 오브젝트의 소유자는 해당 커널 오브젝트를 생성한 프로세스이다.
   3. 커널 오브젝트의 생성 및 관리는 사용자 프로세스에 의해서 이뤄진다.
   4. 운영체제가 생성 및 관리하는 리소스의 종류에 상관없이 커널 오브젝트 데이터 블록 구조는 완전히 동일하다.
2. 오늘날에는 대부분의 운영체제가 운영체제 레벨에서 쓰레드를 지원한다. 이러한 상황을 근거로 하여 다음 설명 중에서 옳지 않은 것을 모두 골라보자.
   1. Main 함수를 호출하는 것도 쓰레드이다.
   2. 프로세스가 쓰레드를 생성하지 않으면, 프로세스 내에는 쓰레드가 하나도 존재하지 않게 된다.
   3. 멀티쓰레드 모델은 프로세스 내에서 추가로 쓰레드를 생성하는 프로그램의 유형을 의미한다.
   4. 단일쓰레드 모델은 프로세스 내에서 추가로 딱 하나의 쓰레드만 추가로 생성하는 프로그램 유형이다.
3. 윈도우의 쓰레드를 메모리 공간에서 완전히 소멸시키는 방법과 리눅스의 쓰레드를 메모리 공간에서 완전히 소멸시키는 방법의 차이점을 비교 설명해보자.

윈도우에서는 쓰레드에 의해서 처음 호출된 또는 쓰레드의 Main 함수가 반환하는 시점에 메모리에서 완전히 소멸된다. 리눅스에서는 쓰레드의 반환이 발생하더라도, join( ) 함수이 호출되어 부모 스레드는 return 하더라도 자식 스레드가 종료 될 때까지 블러킹된다.

1. 커널 오브젝트, 쓰레드, 그리고 핸들의 관계를 쓰레드가 생성되는 상황을 이용해서 설명해보자.

프로세스 내에서 쓰레드를 생성하였을 때, 운영체제는 쓰레드를 관리하기 위해 쓰레드 타입과 관련된 리소스 커널 오브젝트를 생성하여 쓰레드의 상태를 기록한다. 이 때, 프로세스(프로그램)는 핸들을 통하여 리소스 커널 오브젝트에 접근하여 리소스 상태를 변경할 수 있다.

1. 커널 오브젝트와 관련된 다음 문장들 중에서 말하는 바가 옳으면 O, 틀리면 X를 표시하자.

* 커널 오브젝트는 signaled 상태와 Non-signaled 상태 중 하나의 상태가 된다. ( O )
* 커널 오브젝트가 signaled 상태가 되어야 하는 시점에 프로그래머는 직접 커널 오브젝트의 상태를 signaled 상태로 변경해야 한다. ( X )
* 쓰레드의 커널 오브젝트는 쓰레드가 실행 중일 때 signaled 상태에 있다가, 쓰레드가 종료되면 non-signaled 상태가 된다. ( X )

1. ‘auto-reset 모드’ 커널 오브젝트와 ‘manual-reset 모드’ 커널 오브젝트에 대해서 설명하여라. 커널 오브젝트는 어떠한 특징적 차이로 둘 중 하나로 나뉘게 되는가?

이벤트 발생에 의하여 WaitForSingleObject( )/WaitForMultipleObject( ) 함수가 signaled를 반환하였을 때, object의 status를 자동적으로 non-signaled 상태로 바뀌는 커널 오브젝트를 ‘auto-reset 모드’ 커널 오브젝트라 하며,

사용자가 직접 status를 바꿔주어야 하는 오브젝트를 ‘manual-reset 모드’ 커널 오브젝트라고 한다.