

! 뮤텍스와 세마포어의 차이를 설명해주세요!

뮤텍스: 한 스레드, 프로세스에 의해 소유될 수 있는 리소스를 기반으로 한 상호배제 기법.

세마포어: Signaling mechanism. 현재 공유자원에 접근할 수 있는 스레드, 프로세스의 수를 나타내는 값을 두어 상호배제를 달성하는 기법.

차이: 세마포어는 공유자원에 세마포어의 변수만큼의 프로세스(또는 스레드)가 접근할 수 있습니다. 반면에 뮤텍스는 오직 1개의 프로세스(또는 스레드)만 접근할 수 있습니다.

현재 수행 중인 프로세스가 아닌 다른 프로세스가 세마포어를 해제할 수 있습니다. 하지만 뮤텍스는 락을 획득한 프로세스가 반드시 그 락을 해제해야 합니다.

2. 스케줄러란 무엇이고 단기/중기/장기 ~

스케줄러

프로세스를 스케줄링하기 위한 Queue에는 세 가지 종류가 존재한다.

- Job Queue : 현재 시스템 내에 있는 모든 프로세스의 집합
- Ready Queue : 현재 메모리 내에 있으면서 CPU를 잡아서 실행되기를 기다리는 프로세스의 집합
- Device Queue : Device I/O 작업을 대기하고 있는 프로세스의 집합

각각의 Queue에 프로세스들을 넣고 빼주는 스케줄러에도 크게 세 가지 종류가 존재한다.

장기스케줄러(Long-term scheduler or job scheduler)

메모리는 한정되어 있는데 많은 프로세스들이 한꺼번에 메모리에 올라올 경우, 대용량 메모리(일반적으로 디스크)에 임시로 저장된다. 이 pool에 저장되어 있는 프로세스 중 어떤 프로세스에 메모리를 할당하여 ready queue로 보낼지 결정하는 역할을 한다.

- 메모리와 디스크 사이의 스케줄링을 담당.
- 프로세스에 memory(및 각종 리소스)를 할당(admit)
- degree of Multiprogramming 제어
메모리에 여러 프로그램이 올라가는 것) 몇 개의 프로그램이 올라갈 것인지를 제어
- 프로세스의 상태
new -> ready(in memory)

cf) 메모리에 프로그램이 너무 많이 올라가도, 너무 적게 올라가도 성능이 좋지 않은 것이다. 참고로 time sharing system에서는 장기 스케줄러가 없다. 그냥 곧바로 메모리에 올라가 ready 상태가 된다.

단기스케줄러(Short-term scheduler or CPU scheduler)

- CPU와 메모리 사이의 스케줄링을 담당.
- Ready Queue에 존재하는 프로세스 중 어떤 프로세스를 running 시킬지 결정.
- 프로세스에 CPU를 할당(scheduler dispatch)
- 프로세스의 상태
ready -> running -> waiting -> ready

중기스케줄러(Medium-term scheduler or Swapper)

- 여유 공간 마련을 위해 프로세스를 통째로 메모리에서 디스크로 쫓아냄 (swapping)
- 프로세스에게서 memory를 deallocate
- degree of Multiprogramming 제어
- 현 시스템에서 메모리에 너무 많은 프로그램이 동시에 올라가는 것을 조절하는 스케줄러.
- 프로세스의 상태
ready -> suspended

Process state - suspended

Suspended(stopped) : 외부적인 이유로 프로세스의 수행이 정지된 상태로 메모리에서 내려간 상태를 의미한다. 프로세스 전부 디스크로 swap out 된다. blocked 상태는 다른 I/O 작업을 기다리는 상태이기 때문에 스스로 ready state로 돌아갈 수 있지만 이 상태는 외부적인 이유로 suspending 되었기 때문에 스스로 돌아갈 수 없다.

스케줄러란?

OS 내에서 프로세스에게 CPU 위에서 실행될 수 있도록 제어해주는 프로그램.

스케줄러는 크게 선점형(=배정권있는)과 비선점형으로 나뉘어 줍니다.

선점형은 현재 실행중인 프로세스가 있어도 다른 프로세스가 밑에나어 실행할 수 있도록 하는 스케줄러이고, 비선점형은 현재 실행중인 스케줄러가 있으면 해당 프로세스가 종료될때까지 다른 프로세스는 실행할 수 없는 스케줄러입니다.