

Linux Scheduler Virtual Machine

개요

- 리눅스 스케줄러 가상 컴퓨터는 스케줄러 코드와 함께 사용할 수 있는 static C 라이브러리입니다. 이 가상 컴퓨터는 리눅스 커널이 IO 루틴과 프로세스 생성/제거를 처리하는 코드를 단순화한 것입니다. 더불어 리눅스 커널의 기능을 흉내내기 위해 가상 컴퓨터도 스케줄러의 디버그 환경을 제공합니다.
- 가상 컴퓨터는 실제 시간에 맞추어 동작하지 않습니다. 매우 천천히 동작하므로 결과를 직접 확인하며 문제가 있는 부분을 찾을 수 있습니다. 가상 컴퓨터의 동작 속도는 임의로 조절할 수 있습니다.

시작하기

- 스케줄러 코드와 가상 컴퓨터를 함께 컴파일하면 실행파일이 생성됩니다. 이 실행파일을 사용하기 위해 가상 컴퓨터 프로필을 함께 제공해야 합니다. 프로필에는 프로세스를 언제, 어떻게 생성해야 하는지, 어떤 타입의 프로세스를 생성할 것인지, 어떻게 동작해야 하는지 등이 작성되어 있으며, 스크립트 언어를 사용하여 만들 수 있습니다. 더 자세한 내용은 **프로필 스크립트** 부분을 참조하시기 바랍니다.

가상 컴퓨터의 출력문

- 프로필과 함께 가상 컴퓨터를 실행하면, 가상 컴퓨터가 여러 줄을 출력할 것입니다. 이 출력문은 가상 컴퓨터가 무엇을 하고 있는지를 의미하므로, 작성한 스케줄러의 수정할 때 유용할 것입니다. 대부분의 내용은 그 문장 그대로의 상황을 의미하지만, 아래와 같은 출력문은 별도의 이해가 필요할 것입니다.

– 샵이 3개 연달아 나오면 특별한 이벤트가 발생한 것입니다. 특별한 이벤트로는 프로세스 생성과 소멸, IO 이벤트, 가상머신의 시작과 종료 등이 있습니다.

() – 소괄호로 둘러 쌓인 문장은 프로세스에 대한 정보입니다. 소괄호 안에는 이름, 콜론, 숫자가 적혀 있습니다. 이름은 프로세스의 이름을 의미하며, 중복된 이름이 나올 수 있습니다. 숫자는 프로세스의 아이디이며 고유 값입니다. 일부 문장에는 프로세스 정보가 여러 개 나오며, 각 프로세스 정보를 콜론으로 구분합니다. 앞의 있는 프로세스는 부모 프로세스이고, 뒤에 있는 프로세스는 자식 프로세스입니다. 이후에는 슬래시로 구분된 숫자들이 나옵니다. 첫 번째 숫자는 프로세스에게 남아있는 clock tick입니다. 즉, 프로세스가 종료될 때까지 필요한 time-slice를 의미합니다. 두 번째 숫자는 가상 컴퓨터의 현재 system

clock입니다. system clock은 가상 컴퓨터가 구동되고 얼마만큼의 시간(millisecond)이 지났는지를 의미합니다.

프로필 스크립트

- 가상 컴퓨터 프로필에는 프로세스의 정보가 나열되어 있으며, 간단한 스크립트 언어로 작성되어 있습니다. 각 프로세스는 여러 속성을 가지고 있으며 이에 따라 언제 실행되고, 어떻게 동작하는지, 자식 프로세스는 몇 개나 생성하는지, 언제 종료되는지 결정됩니다. 프로필에는 맨 먼저 프로필 헤더가 나오며, 그 다음 프로세스 블록들이 나오게 됩니다.

헤더 블록

- 헤더 블록은 가상 컴퓨터가 어떻게 동작할지에 대한 정보를 담고 있으며 3가지 명령으로 구성됩니다. 헤더 블록은 항상 프로필의 최상단에 위치해야 합니다.

헤더 블록 명령어

명령어	값	설명
#CYCLE_TIME	1 ~ 10000	이 값은 가상 컴퓨터의 clock tick 간 delay를 의미합니다. 작은 값일수록 가상 컴퓨터가 빠르게 동작합니다.
#SEED	0이 아닌 정수	이 값은 랜덤 넘버 생성기의 seed를 의미합니다. 이 값을 바꾸면 가상 컴퓨터의 동작이 달라집니다. 즉, 컴퓨터가 달라지더라도 이 값이 같으면 동일하게 동작합니다.
#ENDTIME	0이 아닌 정수	이 값은 가상 컴퓨터가 동작하고 몇 millisecond 후에 모든 프로세스에게 KILL 명령을 보낼지를 의미합니다.

프로세스 블록

- 프로세스 블록은 프로세스에 대한 속성들로 구성되어 있습니다. 몇 개의 프로세스를 작성하든 상관없지만, 프로세스의 개수가 너무 많아지면 가상 컴퓨터가 종료될 때까지 많은 시간이 필요해질 것입니다.
- I/O bound 는 I/O 작업을 하는데 많은 시간을 소모하는 프로세스를 의미하며, CPU bound 는 I/O 요청은 적게 받고, 대부분 CPU 를 사용해 작업하는 프로세스를 의미합니다. I/O bound 프로세스는 입력을 기다리는데 대부분의 시간을 사용하고 CPU 사용 시간은

짧기 때문에, 일반적으로 I/O bound 프로세스의 우선순위가 CPU bound 프로세스보다 높습니다.

프로세스 블록 명령어

명령어	필수	값	설명
#NEWPROCESS	O	없음	프로세스 블록의 시작을 의미합니다.
#TYPE	O	INTERACTIVE 또는 NONINTERACTIVE	프로세스의 종류를 의미합니다. INTERACTIVE 프로세스는 I/O bound이며, NONINTERACTIVE는 CPU bound입니다.
#NAME	O	영어+숫자. 공백 사용 불가	프로세스의 이름입니다.
#SPAWNTIME	O	0이 아닌 양의 정수	프로세스의 생성시간을 밀리초로 표현한 것입니다. 참고로 프로세스는 스케줄링을 기다려야하기 때문에, 여기 적힌 시간에 정확히 생성되고 실행되지는 않습니다.
#KILLTIME	X	0이 아닌 양의 정수	프로세스가 KILL 메시지를 받은 시간을 의미합니다. 명시하지 않으면 가상 컴퓨터가 종료될 때 KILL 메시지를 받게 됩니다.
#NICE	X	-19 ~ 20	프로세스의 nice 값(우선 순위)입니다. 명시하지 않으면 가상 컴퓨터의 기본 nice 값으로 설정됩니다. 우선 순위를 사용하는 스케줄러에서만 사용됩니다.
#SPAWN	X	없음	Spawn 블록의 시작을 의미합니다. 자식 프로세스는 spawn 블록 안에 정의되어야 합니다. 여러 개의 자식 프로세스를 나열할 수 있으며, spawn 블록이 중첩될 수도 있습니다.
#ENDSPAWN	X	없음	Spawn 블록의 끝을 의미합니다.
#ENDPROCESS	O	없음	프로세스 블록의 끝을 의미합니다.

프로세스 블록 레이아웃

- 프로세스 블록은 항상 #NEWPROCESS 명령으로 시작해서 #ENDPROCESS 명령으로 끝나야 합니다. 각 프로세스 spawn 블록을 최대 하나까지 가질 수 있지만, 그 안에 여러 개의 프로세스 블록을 넣거나, spawn 블록을 중첩하여 사용할 수 있습니다. #TYPE과 #NAME 명령은 항상 #NEWPROCESS 바로 다음에 나와야 합니다.