

Semaphores

2019-11730 신현지

실험 조건

badcnt와 goodcnt 모두 인자를 10000으로 설정하고 10번씩 수행해 평균을 내었다.

컴퓨터 사양

CPU

- Architecture: x86_64
- Model name: i5-12600KF
- CPU(s): 16
- Thread(s) per core: 2
- Core(s) per socket: 10
- CPU max MHz: 4900.0000

RAM

- Total: 16 GB
- Swap: 8 GB

SSD: 953.9 GB

GPU: GeForce RTX 3060 Ti Lite Hash Rate

OS

- Linux
- Kernel version: 5.19.0-41-generic

수행 결과

- badcnt.c : 426 μ s, 모두 BOOM이 출력됨
- goodcnt.c : 5637.6 μ s, 모두 OK가 출력됨

badcnt.c 에서 모두 "BOOM"이 출력된 이유는 다음과 같다. 두 개 이상의 스레드가 동일한 cnt 변수를 변경하려고 하는 상황에서, 어떤 스레드가 먼저 실행되거나 어떤 스레드가 다른 스레드 사이에 끼어들어 실행될지 그 순서가 일정하지 않기 때문이다. 이때 thread 1의 Store 명령보다 thread 2의 Load 명령이 먼저 실행될 경우 문제가 발생한다.

이를 해결하기 위한 한 가지 방법은 mutual exclusion을 사용하는 것이다. goodcnt.c에서는 semaphores를 사용하여 cnt 변수에 대한 동시 접근을 제어하고 있다. 이를 통해 한 번에 하나의 스레드만이 cnt 변수를 증가시키는 것이 보장되므로, 이 프로그램에서는 "OK"가 출력된다. 그러나 P 연산과 V 연산을 사용하면 오버헤드가 발생하여, goodcnt.c는 badcnt.c에 비해 2배 이상의 실행 시간이 소요된다. 이 실험 결과에서는 13배 정도의 실행시간이 소요되었다.

Appendix: 10회의 수행 결과 시간 표

	badcnt	goodcnt
1	445	5112
2	376	5487
3	443	5377
4	394	5569
5	426	5515
6	402	6125
7	419	5633
8	419	5493
9	449	5774
10	487	6291
평균	426	5637.6

단위는 μs 이다.