



제 3 문. 두 개의 프로세스에 대한 임계구역 문제를 소프트웨어적으로 해결하고자 한다.  
다음 물음에 답하시오. (총 20점)

- 1) 임계구역(critical section) 문제를 소프트웨어적으로 해결하기 위하여 만족시켜야 하는 세 가지 조건을 나열하시오. (4점)
- 2) 첫 번째 시도로 다음과 같이 구현하였다. 이 코드는 두 프로세스 중 한 프로세스는 lock\_1(0)으로, 다른 프로세스는 lock\_1(1)로 호출하여 동시에 수행된다. 이것이 임계구역 문제를 해결하는지에 대해 설명하시오. (8점)

```
// 초기값
boolean b[2] = {false, false};
int turn = 0;
. . .

void lock_1(int id) {
    do {
        while(turn != id) { /* 대기 */ }
        . . . // critical section
        turn = 1-id;
        . . . // remainder
    } while(true);
}
```

- 3) 두 번째 시도로 다음과 같이 구현하였다. 두 프로세스가 각각 lock\_2(0)과 lock\_2(1)로 호출하여 동시에 수행된다고 했을 때, 이것이 임계구역 문제를 해결하는지에 대해 설명하시오. (8점)

```
// 초기값
boolean b[2] = {false, false};
int turn = 0;
. . .

void lock_2(int id) {
    while(true) {
        b[id] = true;
        while(turn != id) {
            while(b[1-id]) { /* 대기 */ }
            turn = id;
        }
        . . . // critical section
        b[id] = false;
        . . . // remainder
    }
}
```

제 4 문. 최단 마감 우선(Earliest Deadline First) 스케줄링과 비율 단조(Rate-Monotonic) 스케줄링을 사용하는 실시간 시스템에 주기를 갖는 N개의 태스크가 존재한다고 가정한다. 이 시스템의 각 태스크  $\tau_i$ 는  $(C_i, T_i)$ 로 모델링된다고 가정하고, 다음 표에서와 같은 태스크들이 주어질 때, 물음에 답하시오. (단,  $C_i$ 는  $\tau_i$ 의 실행시간을 의미하고  $T_i$ 는  $\tau_i$ 의 주기를 의미하며, 사용률은 소수점 이하 넷째 자리에서 반올림한다) (총 20점)

	실행시간 (ms)	주기 (ms)
태스크 1	10	50
태스크 2	15	90
태스크 3	3	30
태스크 4	6	36
태스크 5	20	180
태스크 6	3	45
태스크 7	2	20
태스크 8	4	60

- 주어진 태스크들에 대해 최단 마감 우선 스케줄링이 가능한지 설명하시오. (6점)
- 비율 단조 스케줄링에서 각 태스크를 우선순위가 높은 것부터 순서대로 표시하시오. (6점)
- 다음 표는 비율 단조 스케줄링의 사용률 상한을 나타낸다.

태스크 개수	사용률 상한
1	1
2	0.828
3	0.779
4	0.756
5	0.743
6	0.734
7	0.728
8	0.724

비율 단조 스케줄링 기법에서 모든 태스크들의 주기(데드라인) 내 수행을 보장하는 충분조건을 기술하고, 이를 만족하는 최대 태스크 집합을 구하시오. (8점)

제 5 문. 리눅스 시스템에서 backdoor 파일과 backdoor2 파일의 정보가 다음과 같다. 물음에 답하시오. (총 20점)

```
$ ls -l backdoor*
-rwsr-xr-x 1 root root 12003 April 3 22:00 backdoor
-rwxr-xr-x 1 root root 12010 April 5 12:30 backdoor2
```

- 리눅스 시스템의 프로세스가 부여받는 RUID(Real User ID), EUID(Effective User ID), RGID(Real Group ID), EGID(Effective Group ID)에 대하여 설명하시오. (4점)
- 리눅스 시스템에서 SetUID 비트를 사용하는 이유를 설명하시오. (4점)
- 사용자 kim의 계정 정보와 그룹 정보가 아래와 같다. 사용자 kim이 backdoor와 backdoor2를 각각 실행했을 때, 해당 프로세스의 RUID, EUID, RGID, EGID 값을 모두 구하시오. (6점)

```
/etc/passwd 파일의 사용자 계정 정보
kim:x:1000:1000:kim:/home/kim:/bin/bash
/etc/group 파일의 사용자 그룹 정보
kim:x:1000:
```

- backdoor와 backdoor2는 매개변수로 파일 이름을 받아 해당 파일의 내용을 화면에 출력하는 프로그램이며, /etc/shadow 파일의 정보는 다음과 같다.

```
-rw-r----- 1 root root 1320 May 4 18:00 shadow

$ ./backdoor /etc/shadow
$ ./backdoor2 /etc/shadow
```

일반 계정 kim으로 로그인한 사용자가 다음과 같은 명령을 각각 실행하였을 때, 실행 결과를 설명하시오. (단, 사용자 kim의 현재 디렉터리에 두 프로그램이 저장되어 있다고 가정한다) (6점)

## 인사혁신처 시험출제과장