

5주차 과제 - 4장. 스레드와 멀티태스킹

1. 올해 넥슨의 채용계약형 인턴 선발에서 나온 질문이다. ‘프로세스의 스레드의 차이점이 무엇인지 설명해보라’ 이 질문에 대해 3장과 4장을 공부한 지금 차이점을 10줄로 설명하라. 다양한 관점에서 비교하고 진지하게 답을 써보라. 스레드가 왜 프로세스보다 나은 실행 단위인지에 대해 잘 드러나도록 설명하라.
2. 스레드란 무엇인지에 대한 질문을 받았다고 생각하고 스레드를 정의해보라.
3. 스레드의 컨텍스트란 어떤 정보를 말하는가? 그리고 어디에 저장되는가?
4. 스레드의 주소 공간은 어디에 만들어지는가? 하나의 스레드가 실행되기 위해 주어지는 주소 공간은 총 6개의 영역으로 나뉘어진다. 이 6가지가 무엇인지 설명하라. 이 중에서 다른 스레드와 공유하는 공간은 무엇인가?
5. 프로세스 1에 속한 스레드에서 프로세스2가 속한 스레드로 컨텍스트 스위칭이 일어날 때, 같은 프로세스에 속한 스레드 스위칭의 경우보다 추가적으로 더 필요한 작업은 무엇인가?
6. 우리가 멀티스레딩 프로그램을 작성할 때 사용자 레벨 스레드로 만들면 어떤 장점이 있으며, 커널 레벨 스레드로 만들면 어떤 장점이 있는가?
7. 내가 작성한 함수는 사용자 영역에 주소 공간이 있다. 이 함수가 커널 레벨 스레드로 실행될 수 있는가? 없다면 이유를, 있다면 실행되기 위해 개발자가 취해야할 필요한 조치는 무엇인가?
8. N:1 매핑은 멀티스레드 응용프로그램에서 만든 스레드 중 한 개가 파일 입출력으로 블록 상태가 되면 응용프로그램 내의 모든 스레드가 스케줄되지 못한다. 이 과정을 그림으로 그리고 간단히 설명하라.
9. 1:1 매핑이 오늘날 컴퓨터 시스템에서 많이 사용되는 이유가 무엇인지 설명하라.
10. 교재의 복합 문제 1의 멀티스레드 응용프로그램을 작성하라. 과제 제출 PDF에 프로그램과 실행 결과를 함께 보여라. 문제는 다음과 같다.

그림 4-5의 맛보기 프로그램을 참고하여, 리눅스에서 다음 5가지 조건에 부합하는 멀티스레드 C 응용프로그램을 작성하라. 4개의 스레드를 활용하여 1에서 40000까지의 합을 구하여 출력하라.

- (1) pthread 라이브러리를 이용하여 작성하라.
- (2) 응용프로그램에 전역 변수 int sum[4]을 선언하라.
- (3) 스레드로 실행할 함수의 이름을 runner로 하라.
- (4) main()에서 4개의 스레드를 연속적으로 생성하여 다음과 같이 이용하라. 4개의 스레드는 동시에 실행된다.

스레드 1 : 1~10000까지의 합을 구하고 sum[0]에 저장

```
pthread_create(&, __, runner, "1"); // 1에서 10000까지 합 구하기
```

스레드 2 : 10001~20000까지의 합을 구하고 sum[1]에 저장

```
pthread_create(&, __, runner, "10001"); // 10001에서 20000까지 합 구하기
```

스레드 3 : 20001~30000까지의 합을 구하고 sum[2]에 저장

```
pthread_create(&, __, runner, "20001"); // 20001에서 30000까지 합 구하기
```

스레드 4 : 30001~40000까지의 합을 구하고 sum[3]에 저장

```
pthread_create(&, __, runner, "30001"); // 30001에서 40000까지 합 구하기
```

(5) main 스레드는 4개의 스레드가 모두 종료하기를 기다린 후 sum[] 배열의 값을 모두 합쳐 그 결과를 화면에 출력하라. 다음은 이 프로그램의 이름이 prac4_1.c라고 할 때 컴파일 과정과 실행 결과를 보여준다.

```
$ gcc -o prac4_1 prac4_1.c -lpthread
$ ./prac4_1
1에서 40000까지 4개의 스레드가 계산한 총 합은 800020000
$
```

11. 교재의 복합 문제 2의 멀티스레드 응용프로그램을 작성하라. 과제 제출 PDF에 프로그램과 실행 결과를 함께 보여라. 문제는 다음과 같다.

앞의 문제를 수정하여 리눅스에서 다음 5가지 조건에 부합하는 멀티스레드 C 응용프로그램을 작성하라. 4개의 스레드를 활용하여 1에서 40000까지의 합을 구하여 출력하라.

- (1) pthread 라이브러리를 이용하여 작성하라.
- (2) 응용프로그램에 전역 변수 int sum을 선언하라.
- (3) 스레드로 실행할 함수의 이름을 runner로 하라.
- (4) main()에서 4개의 스레드를 연속적으로 생성하여 다음과 같이 이용하라. 4개의 스레드는 동시에 실행된다.

스레드 1 : 1~10000까지의 합을 구하고 sum에 저장

`pthread_create(&, &, runner, "1");` // 1에서 10000까지 합 구하기

스레드 2 : 10001~20000까지의 합을 구하고 sum에 저장

`pthread_create(&, &, runner, "10001");` // 10001에서 20000까지 합 구하기

스레드 3 : 20001~30000까지의 합을 구하고 sum에 저장

`pthread_create(&, &, runner, "20001");` // 20001에서 30000까지 합 구하기

스레드 4 : 30001~40000까지의 합을 구하고 sum에 저장

`pthread_create(&, &, runner, "30001");` // 30001에서 40000까지 합 구하기

(5) main 스레드는 4개의 스레드가 모두 종료하기를 기다린 후 sum 값을 화면에 출력하라. 다음은 이 프로그램의 이름이 prac4_2.c라고 할 때 컴파일 과정과 실행 결과를 보여준다.

```
$ gcc -o prac4_2 prac4_2.c -lpthread
$ ./prac4_2
1에서 40000까지 4개의 스레드가 합친 sum 변수의 값은 583862451
$ ./prac4_2
1에서 40000까지 4개의 스레드가 합친 sum 변수의 값은 640209990
$ ./prac4_2
1에서 40000까지 4개의 스레드가 합친 sum 변수의 값은 500849160
$
```

1에서 40000까지의 더한 합은 문제 1의 결과에 따라 800020000인데, 화면에는 800020000가 출력되지 않는다. 여러 번 실행해도 계속 다른 값이 출력된다. 그 이유는 무엇인지 나름대로 설명해보라. 참고로 멀티스레딩에 있어 발생할 수 있는 이 문제를 해결하기 위해, 5장에서는 스레드 동기화 기법을 다룬다.