

운영체제 과제 #1

1. 목적

간단한 Linux 커널 모듈과 연결형 리스트 커널 모듈을 생성하고 Linux 커널에 추가 및 삭제함으로써 운영체제의 개념을 이해하도록 함

2. 과제명: Linux 커널 모듈 생성, 추가 및 삭제

3. 내용

가. 과제 1-1: 간단한 리눅스 커널 모듈

1) 커널모듈: simple.c

```
#include <linux/init.h>
#include <linux/module.h>
#include <linux/kernel.h>

/* This function is called when the module is loaded. */
int simple_init(void)
{
    printk(KERN_INFO "Loading Module\n");

    return 0;
}

/* This function is called when the module is removed. */
void simple_exit(void) {
    printk(KERN_INFO "Removing Module\n");
}

/* Macros for registering module entry and exit points. */
module_init( simple_init );
module_exit( simple_exit );

MODULE_LICENSE("GPL");
MODULE_DESCRIPTION("Simple Module");
MODULE_AUTHOR("SGG");
```

- 2) 각 문장별로 한글 주석을 달 것 (구체적으로 의미를 설명)
- 3) 커널모듈 simple.c를 제공되는 makefile을 사용하여 컴파일
- 4) 커널모듈을 로딩 및 삭제

- sudo insmod simple.ko

- sudo rmmod simple

5) 적절히 수행했는지 확인하기 위해 리눅스 CLI(Command Line Interpreter)에서 “dmesg” 명령어를 이용하여 커널 버퍼 내용을 확인

※ 커널 로그 버퍼가 넘칠 수 있으므로 주기적으로 비워주어야 함

- sudo dmesg -c

5) 모든 주요 수행과정의 내용 화면을 캡처할 것

나. 과제 1-2: 연결형 리스트 커널 모듈

1) 과제 1-1을 참고하여 4명의 생년월일을 포함한 연결형 리스트(linked list)를 생성하고, 연결형 리스트를 순회하여 그 내용을 커널 로그 버퍼로 출력하고 확인할 것

2) 구조체 정의

```
struct birthday
{
    int month;
    int day;
    int year;

    struct list_head list;
};
```

※ include 파일: <linux/list.h>

3) 연결형 리스트(linked list)에 요소들(4명)을 입력

```
static LIST_HEAD(birthday_list);
// birthday_list라는 이름의 list_head 오브젝트를 정의하고 초기화하는 매크로
```

```
struct birthday *ptr;
```

```
person_one = kmalloc(sizeof(*person_one), GFP_KERNEL);
person_one->month = 7;
person_one->day = 8;
person_one->year = 1996;
INIT_LIST_HEAD(&person_one->list);
```

```
list_add_tail(&person_one->list, &birthday_list); //linked list에 추가함
```

4) 연결형 리스트를 순회하여 그 내용을 커널 로그 버퍼로 출력

```
struct birthday *ptr;  
list_for_each_entry(ptr, &birthday_list, list) {  
    //내용을 커널 로그 버퍼에 출력하는 부분을 코딩할 것  
}
```

5) 연결형 리스트로부터 요소들을 제거

```
list_for_each_entry_safe(ptr, next, &birthday_list, list) {  
    printk(KERN_INFO "Removing %d %d %d\n", ptr->month,  
        ptr->day, ptr->year);  
    list_del(&ptr->list);  
    kfree(ptr);  
}
```

6) 각 문장별로 한글 주석을 달 것 (구체적으로 의미를 설명)

- 특히, 각 매크로에 대해 상세하게 설명

7) 연결형 리스트가 적절히 구축되었는지 확인하기 위해 “dmesg”를 이용하여 커널 버퍼 내용을 확인

8) 모듈의 삭제 시 연결형 리스트의 요소를 삭제하고 할당 해지 메모리를 커널에 반납

9) 커널이 삭제되면 연결형 리스트가 제거되었는지 확인하기 위해 “dmesg”를 이용하여 커널 버퍼 내용을 확인

10) 모든 주요 수행과정의 내용 화면을 캡처할 것

다. 과제1-3: 과제 1-1과 1-2의 경험을 기반으로 리눅스 운영체제 구조에 대해 상세하게 설명 (장·단점 포함)

4. 고려사항

가. 윈도우상에서 리눅스 가상 머신 사용

- Oracle의 VirtualBox
- Linux Virtual Machine

5. 행정사항

가. 제출기한: 2020년 9월 25일(금요일) 18:00

- 지각 제출 시 80%
- 6시간 경과 시 50%
- 12시간 경과 시 자동 “0”점 처리

나. 제출장소: 아주Bb

다. 제출방법

- 제출내용
 - . 소스코드 파일(c), 실행 파일(ko), 모든 실행결과 화면 캡처 파일
 - . 보고서
 - .. Linux 커널 모듈 생성, 추가 및 삭제 설명서
 - * makefile 내용 설명 포함
 - .. 리눅스 운영체제 구조 설명서
- 모든 제출내용을 1개의 압축파일로 만들 것
- 압축파일 이름은 ‘과제1_이름’ 으로 할 것
 - . ‘과제1-1’, ‘과제1-2’, ‘과제1-3’ 폴더를 만들 것

<과제설명의 끝>