

# 운영체제 및 실습 과제

## -3주차-

분반:01

학번:201404377

이름: 진승언

```

u201404377@u201404377: ~/Desktop
1 #include <linux/init.h>
2 #include <linux/kernel.h>
3 #include <linux/module.h>
4
5 static int num1 = 0;
6 static int num2 = 0;
7
8 int __init init_my_homework(void){
9     printk("Hello module! My classId is %d, studentId is %d",num1,num2);
10    return 0;
11 }
12 void __exit exit_my_homework(void){
13     printk("Bye!\n");
14 }
15
16 module_param(num1, int, 0);
17 module_param(num2, int, 0);
18
19 module_init(init_my_homework);
20 module_exit(exit_my_homework);
21 MODULE_LICENSE("GPL");
22
"module_homework.c" 22L, 478C

```

1. vi 명령어로 module\_homework.c 를 만들어준다. init에는 출력문에 num1, num2가 출력할 수 있게해주고 module\_param으로 매개변수를 입력 받을 수 있게 해준다. module\_param의 괄호안은 매개변수이름, 자료형타입, default 값이 들어간다. 또한 module\_init()은 모듈이 insmod 명령어를 통해 적재될 때 호출되는 함수를 지정하고 module\_exit()는 rmmod 명령어를 통해 커널에서 제거될 때 호출되는 함수를 지정한다. MODULE\_LICENSE는 GPL라이선스로 설정한다.

```

u201404377@u201404377: ~/Desktop
1 obj-m := module_homework.o
2 KDIR := /lib/modules/$(shell uname -r)/build
3 PWD := $(shell pwd)
4
5
6 all:
7     $(MAKE) -C $(KDIR) M=$(PWD) modules
8 clean:
9     $(MAKE) -C $(KDIR) M=$(PWD) clean

```

2. vi 명령어로 Makefile을 만들어 준다. (obj-m을 module\_homework.o로 해준다) \$(shell uname -r)과 \$(shell pwd)로 폴더가 변경되거나 커널 버전이 변경되어도 정상적으로 make를 할 수 있게 해준다

```

u201404377@u201404377:~/Desktop$ make
make -C /lib/modules/4.15.14/build M=/home/u201404377/Desktop modules
make[1]: Entering directory '/usr/src/linux-4.15.14'
  CC [M]  /home/u201404377/Desktop/module_homework.o
Building modules, stage 2.
MODPOST 1 modules
  CC      /home/u201404377/Desktop/module_homework.mod.o
  LD [M]  /home/u201404377/Desktop/module_homework.ko
make[1]: Leaving directory '/usr/src/linux-4.15.14'
u201404377@u201404377:~/Desktop$ ls
Makefile          module_homework.mod.c  modules.order      wakeup
module_homework.c module_homework.mod.o  Module.symvers     wakeup.c
module_homework.ko module_homework.o      module_test.c
u201404377@u201404377:~/Desktop$

```

3. make명령어를 이용해 컴파일 해준다. (ls명령어를 통해 생성 되었는지 확인한다)

```

u201404377@u201404377:~/Desktop$ sudo insmod module_homework.ko num1=1 num2=2014
04377
u201404377@u201404377:~/Desktop$ lsmod | grep 'module_homework'
module_homework      16384  0
u201404377@u201404377:~/Desktop$ modinfo module_homework.ko
filename:
/home/u201404377/Desktop/module_homework.ko
license:
GPL
srcversion:
6F585A554BC7A416AA7108D
depends:
retpoline:
Y
name:
module_homework
vermagic:
4.15.14 SMP mod_unload
parm:
num1:int
parm:
num2:int
u201404377@u201404377:~/Desktop$

```

4. sudo insmod 명령어를 통해 파라미터를 전달해줌과 동시에 module\_homework를 모듈에 적재해준다. lsmod | grep와 modinfo명령어를 통해 적재된 모듈을 확인한다.

```

[ 6604.406551] Hello module! My classId is 1, studentId is 201404377
[ 6626.472353] Bye!
root@u201404377:/home/u201404377/Desktop#

```

6. 모듈을 rmmod 명령어로 제거하고 dmesg명령어로 커널 메시지를 확인 할 수 있다.