

## Linux Device Drivers – Mini project

### **Q1. Write a program to make your own modprobe command. (Duration: 1hour 30 min)**

**Example:** add.c ,avg.c and Makefile

Avg.c file depends on add.c file.

```
$sudo ./mymodprobe avg
```

#### **Pseudo code**

**Step1:** open modules.dep file

modules.dep file location: `"/lib/modules/`uname -r`/modules.dep"`

``uname -r`` field has to be replaced with your current kernel version which can be easily found using the `uname -r` command output.

Hint: programmer to find out kernel version using `uname()` system call.

**Step2:** Read each line in modules.dep file and store in to local buffer.

In local buffer, search module name ("avg" in the given example) as given as the command line argument.

Repeat these steps to remaining lines in modules.dep file till the module name matches.

Hint: use `fgets()` to read line by line.

You can use your own built function or can use any library functions like `strstr()` to find the substring inside the buffer/main string.

**Step3:** Once you find the given filename from the buffer/main string, find the dependency modules that are written after the `:'` symbol.

Eg: `extra/avg.ko: extra/add.ko` //Here the dependency module is add.ko

**Step4:** Extract the names of the corresponding dependency module/modules to a buff using string manipulations and append that module name to the `insmod` command.

Example: suppose buff `--> add.ko`

```
printf( command_buff, "sudo insmod %s", buff ); //now command_buff is --> sudo insmod add.ko
```

**Step5:** Load the given dependency module/modules using 'insmod' command. This command can be executed using `system()` system call.

Example: `system( command_buff );`

**Step6:** Finally load the filename given as argument also using `insmod` command.

```
Example: printf( command_buff, "sudo insmod %s.ko", argv[1] );
        system( command_buff );
```

**Step7: Testing**

```
$ make clean
$ make
$ sudo make install
$ sudo ./mymodprobe avg
```