Operating Systems Project Report

Project Number (01 / 02 / 03):	03
Name:	呂苾瑄
Student ID:	0816057
YouTube link (Format youtube.com/watch?v=[key]):	https://youtu.be/rN-Jh-wFUXI
Date (YYYY-MM-DD):	2021/12/
Names of the files uploaded to E3:	calculator.c, calculatorModule.c, OS_Project03_0816057.pdf
Physical Machine Total RAM (Example: 8.0 GB):	24.0GB
Physical Machine CPU (Example: Intel i7-2600K):	Intel i5-11400F

Checklist	
Yes/No	**Item**
y	**The report name follows the format "OS_ProjectXX_StudentID.pdf".**
y	**The report was uploaded to E3 before the deadline.**
y	**The YouTube video is public, and anyone with the link can watch it.**
y	**The audio of the video has a good volume.**
y	**The pictures in your report and video have a good quality.**
y	**All the questions and exercises were answered inside the report.**
y	**I understand that late submission is late submission, regardless of the time uploaded.**
y	**I understand that any cheating in my report / video / code will not be tolerated.**

Questions

1. static kernel modules: compiled as part of the base kernel and it is available at any time.

dynamic kernel modules: compiled as modules separately and loaded based on user demand.

The other name of dynamic module is Loadable Kernel Modules (LKM).

differences between system calls and dynamic kernel modules:

修改或增加 system call 要在 kernel 的 source file, 編輯 modules 則不用

use different header files, system call use <linux/syscalls.h>, module use <linux/module.h>

使用 module 的時候會呼叫 system call

修改 system call 之後要 recompile the kernel，修改 modules 不需要

2. 新增或修改 system call 是在 kernel 的 source file 中，會增加一些路徑以及 system call 的 id，因此需要 recompile the kernel。Module 是被載入到 kernel 中的，不在 kernel source file 裡面，修改後也不需要 recompile 整個 kernel

3. insmod: install module，rmmod: remove module，modinfo: 顯示 module 的訊息，像是 author, description, license, parameters

sudo insmod dummyModule.ko 後面可以視需求放參數

sudo modinfo dummyModule.ko 會印出 module 的訊息

sudo rmmod dummyModule 移除此 module

4.

a. module_init(函數) ， 定義載入 module 時要調用的函數

b. module_exit(函數)，定義 module 移除時要調用的函數

c. MODULE_LICENSE(“GPL”)，module 的許可證，沒有的話會收到 kernel tainted 的警告，其它能用的許可證有”GPL v2”, “GPL and additional rights”, “Dual BSD/GPL”, “Dual MPL/GPL”, “Proprietary”

d. module_param(name, type, perm)，表示變數的名稱、型態、perm 是存取權限。載入 module 的同時帶參數進去，也是 modinfo 的 parm 會顯示變數的名稱及型態

e. MODULE_PARM_DESC(名稱, 內容描述)，用來描述驅動 module 參數的訊息

5.

a. cat(concatenate): 印出文件內容

b. ls(list files): 印出當前目錄下的內容。ls -l: 除了名稱，也將文件型態、權限、擁有者、文件大小等詳細資料印出來

c. dmesg(display message): 顯示開機訊息。-w: Wait for new messages. -H: Enable human-readable output.

d. lsmod(list modules): 顯示已經載入系統的 modules

e. lsmod | grep: 在已載入的 module 中篩選出符合條件的。grep 表示用關鍵字或是正規表示法篩選

6. module_param 的第三個 parameter 是權限，0644 for root-writable

7. 一開始我的 parameter 放錯了，在 init 時發生錯誤，顯示 Exec format error

8. dummyStudentId, dummySecretValue

9. /sys/module 儲存系統中所有 modules 的訊息

例如可以從/sys/module/(moduleName)/parameters/(parameterName)去修改某 module 的某參數

10.charp: define a module parameter that takes a string

Additional questions

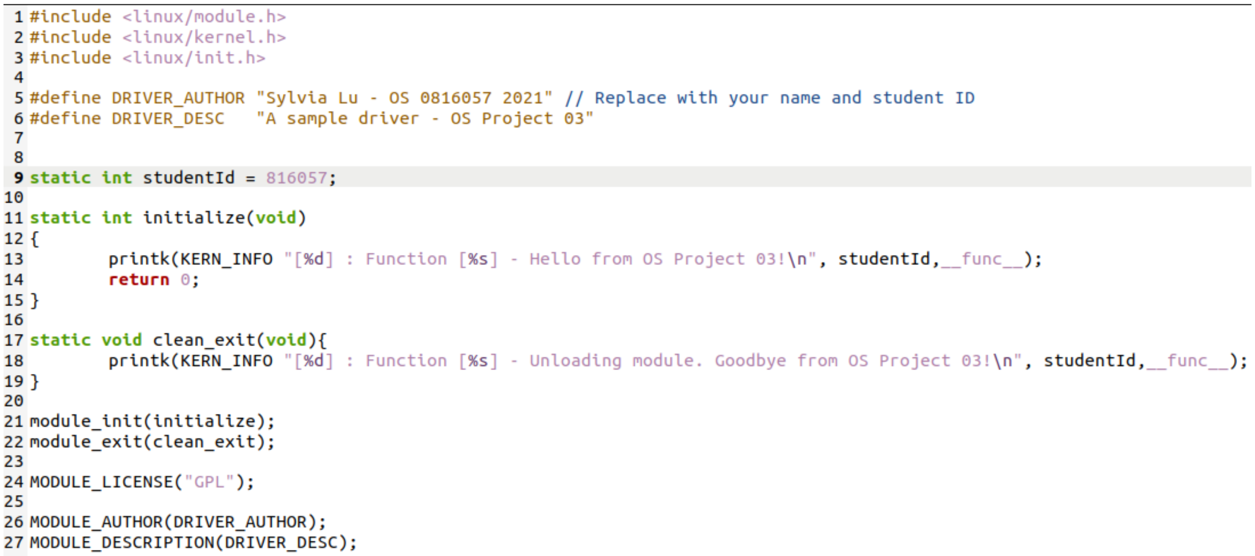
11. Project03. 因為影片的素材程度是最短的，也是最好處理的，有時候影片太長或是過高倍速電腦會卡。Project03 也是在了解 .c file 裡面的含意之後最好完成的一個 project

12. Project01. 一開始對於 Linux 非常不熟悉，虛擬機也不太了解，加上之前的課程凡是有用到虛擬機的總是搞不定，懷疑自己到底能不能完成作業。(不過其實每個 project 的 questions 都不是很好回答)

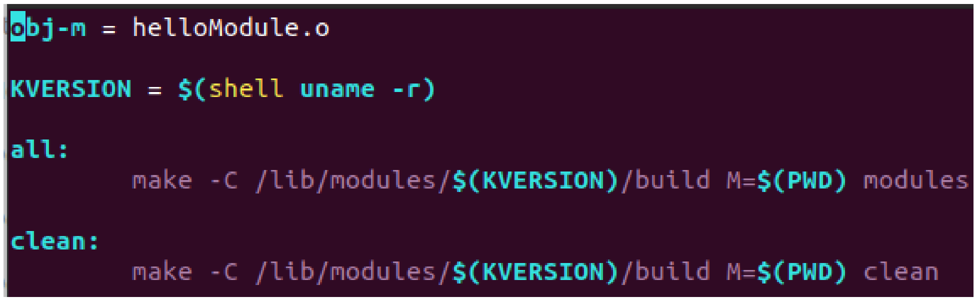
13. 學會一些虛擬機的基本操作，也透過漸進的 project 一步步認識 OS。在這之前我對於 OS 幾乎沒什麼概念，只知道是相對底層的東西，這幾個 project 讓我從 kernel, system call, KGDB, kernel module 一個一個認識，雖然都只接觸一小部分，但也讓我對於這些課本中提到的東西有更多了解。同時也對於影片剪輯有一些了解，如果不是做作業，平時應該也不會去學

14.我目前也不曉得之後會做什麼樣的工作，也許不一定會碰到比較底層的東西，但虛擬機可以用在很多地方，熟悉它的操作我相信會有幫助的。

Screenshot #1

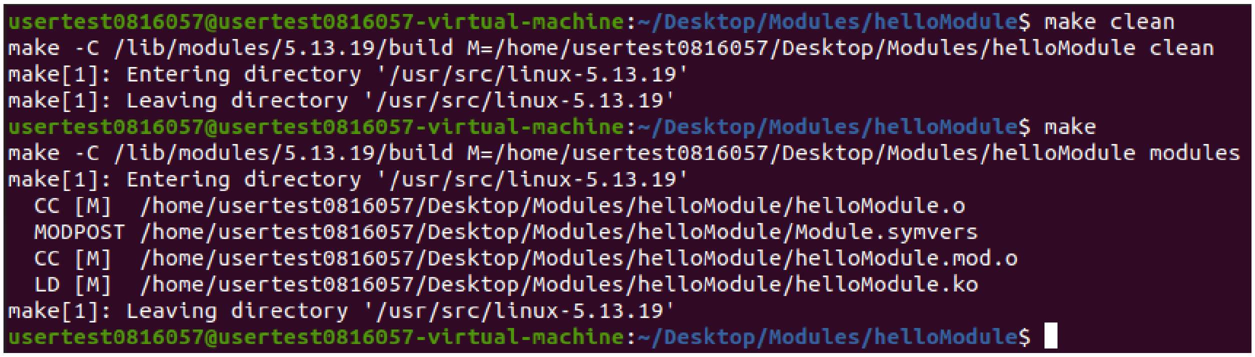


```
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/init.h>

#define DRIVER_AUTHOR "Sylvia Lu - OS 0816057 2021" // Replace with your name and student ID
#define DRIVER_DESC   "A sample driver - OS Project 03"


static int studentId = 816057;

static int initialize(void)
{
        printk(KERN_INFO "[%d] : Function [%s] - Hello from OS Project 03!\n", studentId,__func__);
        return 0;
}

static void clean_exit(void){
        printk(KERN_INFO "[%d] : Function [%s] - Unloading module. Goodbye from OS Project 03!\n", studentId,__func__);
}

module_init(initialize);
module_exit(clean_exit);

MODULE_LICENSE("GPL");

MODULE_AUTHOR(DRIVER_AUTHOR);
MODULE_DESCRIPTION(DRIVER_DESC);
```



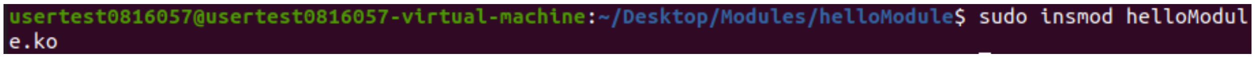
```
obj-m = helloModule.o

KVERSION = $(shell uname -r)

all:
        make -C /lib/modules/$(KVERSION)/build M=$(PWD) modules

clean:
        make -C /lib/modules/$(KVERSION)/build M=$(PWD) clean
```

Screenshot #2

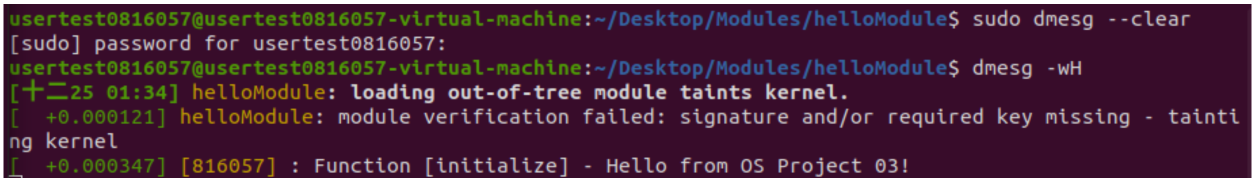


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ make clean
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/helloModule clean
make[1]: Entering directory '/usr/src/linux-5.13.19'
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ make
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/helloModule modules
make[1]: Entering directory '/usr/src/linux-5.13.19'
  CC [M]  /home/usertest0816057/Desktop/Modules/helloModule/helloModule.o
  MODPOST /home/usertest0816057/Desktop/Modules/helloModule/Module.symvers
  CC [M]  /home/usertest0816057/Desktop/Modules/helloModule/helloModule.mod.o
  LD [M]  /home/usertest0816057/Desktop/Modules/helloModule/helloModule.ko
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ 
```

Screenshot #3

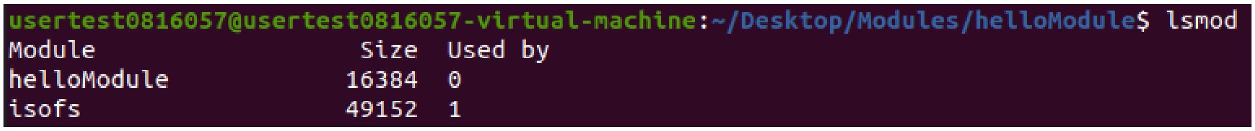


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ sudo insmod helloModul
e.ko
```

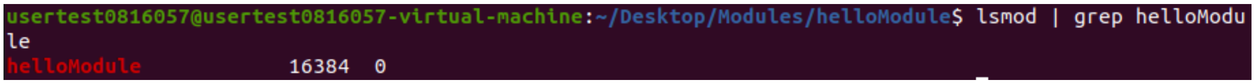


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ sudo dmesg --clear
[sudo] password for usertest0816057:
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ dmesg -wH
[十二25 01:34] helloModule: loading out-of-tree module taints kernel.
[  +0.000121] helloModule: module verification failed: signature and/or required key missing - tainti
ng kernel
[  +0.000347] [816057] : Function [initialize] - Hello from OS Project 03!
```

Screenshot #4

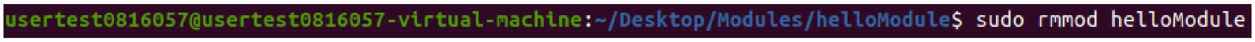


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ lsmod
Module                  Size  Used by
helloModule            16384  0
isofs                  49152  1
```

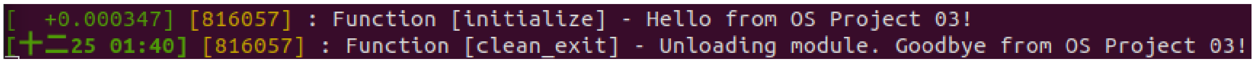


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ lsmod | grep helloModu
le
helloModule            16384  0
```

Screenshot #5

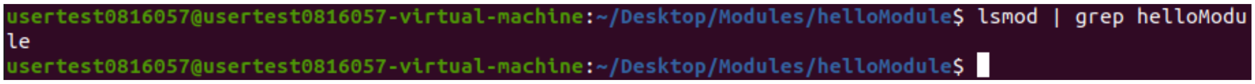


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ sudo rmmod helloModule
```



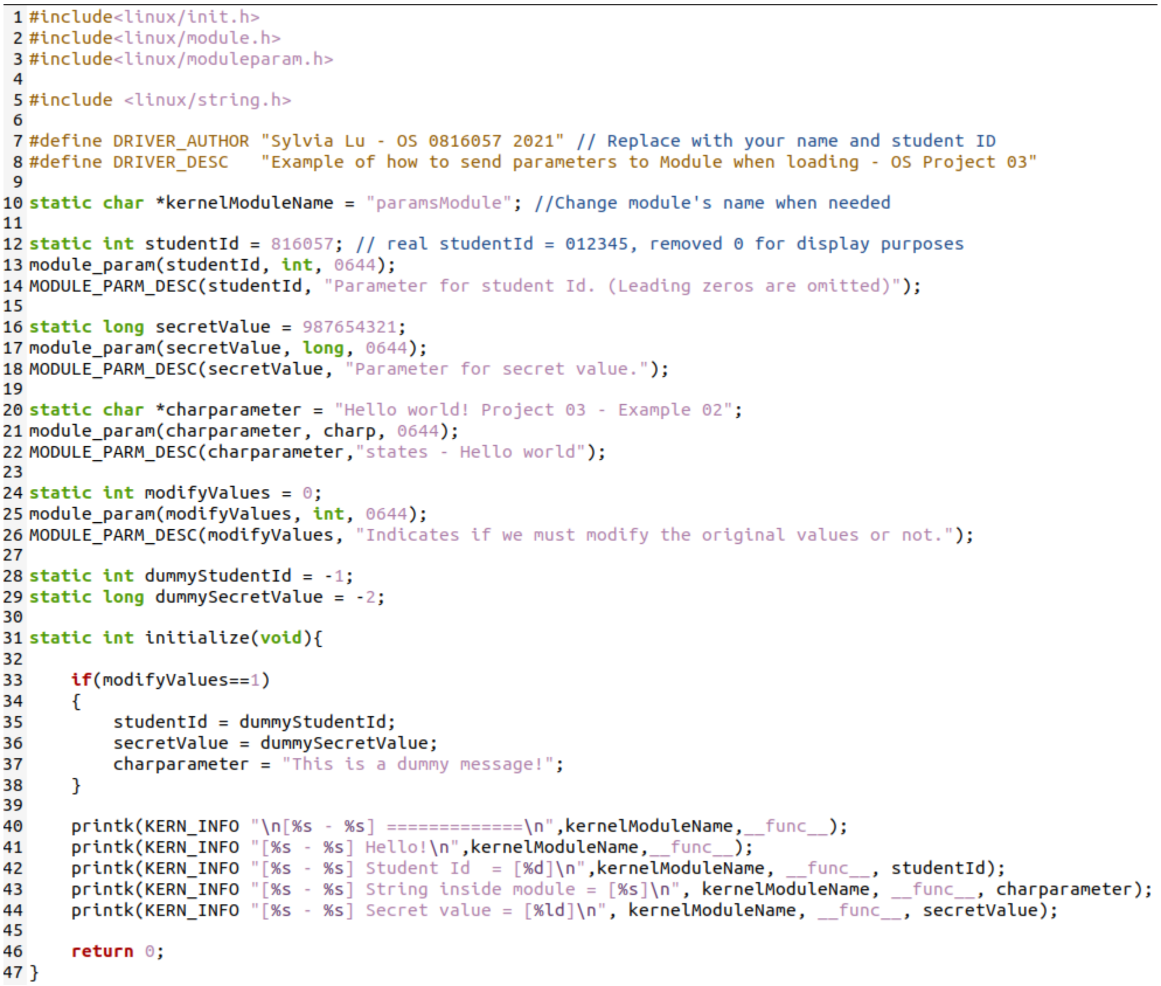
```
[  +0.000347] [816057] : Function [initialize] - Hello from OS Project 03!
[十二25 01:40] [816057] : Function [clean_exit] - Unloading module. Goodbye from OS Project 03!
```

Screenshot #6

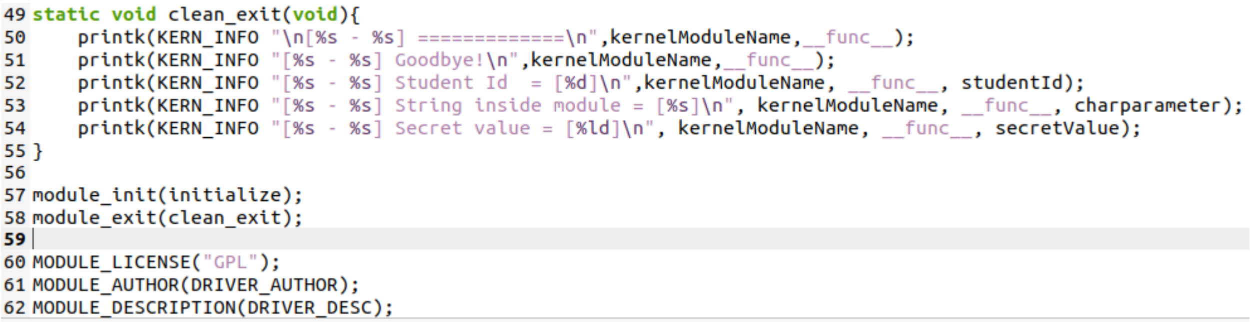


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ lsmod | grep helloModu
le
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/helloModule$ 
```

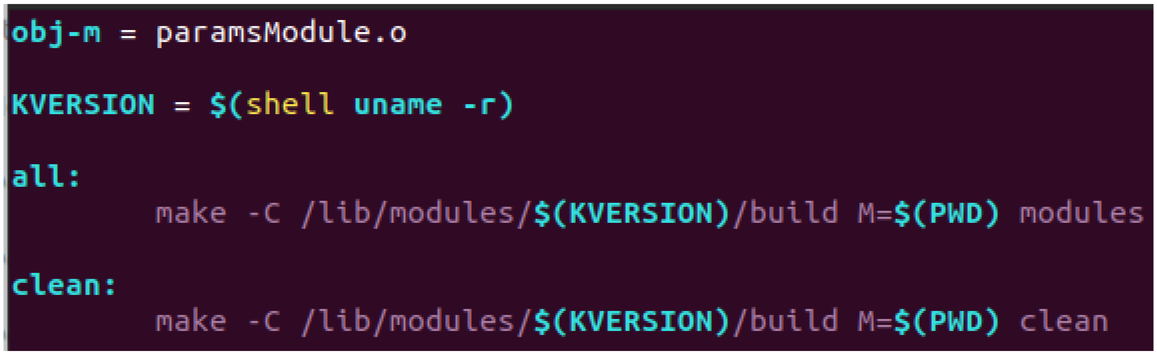
Screenshot #7



```c
#include<linux/init.h>
#include<linux/module.h>
#include<linux/moduleparam.h>

#include <linux/string.h>

#define DRIVER_AUTHOR "Sylvia Lu - OS 0816057 2021" // Replace with your name and student ID
#define DRIVER_DESC   "Example of how to send parameters to Module when loading - OS Project 03"

static char *kernelModuleName = "paramsModule"; //Change module's name when needed

static int studentId = 816057; // real studentId = 012345, removed 0 for display purposes
module_param(studentId, int, 0644);
MODULE_PARM_DESC(studentId, "Parameter for student Id. (Leading zeros are omitted)");

static long secretValue = 987654321;
module_param(secretValue, long, 0644);
MODULE_PARM_DESC(secretValue, "Parameter for secret value.");

static char *charparameter = "Hello world! Project 03 - Example 02";
module_param(charparameter, charp, 0644);
MODULE_PARM_DESC(charparameter,"states - Hello world");

static int modifyValues = 0;
module_param(modifyValues, int, 0644);
MODULE_PARM_DESC(modifyValues, "Indicates if we must modify the original values or not.");

static int dummyStudentId = -1;
static long dummySecretValue = -2;

static int initialize(void){

    if(modifyValues==1)
    {
        studentId = dummyStudentId;
        secretValue = dummySecretValue;
        charparameter = "This is a dummy message!";
    }

    printk(KERN_INFO "\n[%s - %s] =============\n",kernelModuleName,__func__);
    printk(KERN_INFO "[%s - %s] Hello!\n",kernelModuleName,__func__);
    printk(KERN_INFO "[%s - %s] Student Id  = [%d]\n",kernelModuleName, __func__, studentId);
    printk(KERN_INFO "[%s - %s] String inside module = [%s]\n", kernelModuleName, __func__, charparameter);
    printk(KERN_INFO "[%s - %s] Secret value = [%ld]\n", kernelModuleName, __func__, secretValue);

    return 0;
}
```

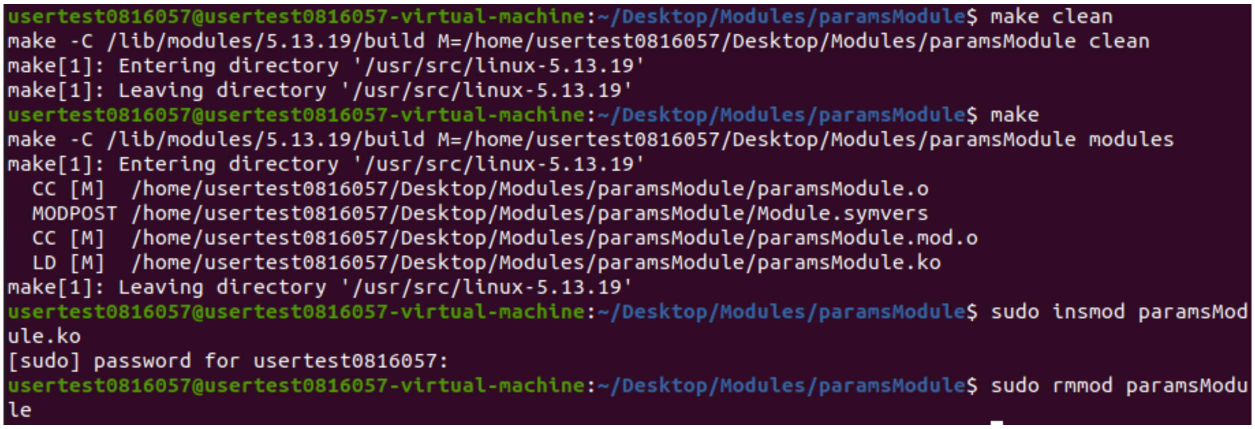


```
static void clean_exit(void){
    printk(KERN_INFO "\n[%s - %s] =============\n",kernelModuleName,__func__);
    printk(KERN_INFO "[%s - %s] Goodbye!\n",kernelModuleName,__func__);
    printk(KERN_INFO "[%s - %s] Student Id  = [%d]\n",kernelModuleName, __func__, studentId);
    printk(KERN_INFO "[%s - %s] String inside module = [%s]\n", kernelModuleName, __func__, charparameter);
    printk(KERN_INFO "[%s - %s] Secret value = [%ld]\n", kernelModuleName, __func__, secretValue);
}

module_init(initialize);
module_exit(clean_exit);

MODULE_LICENSE("GPL");
MODULE_AUTHOR(DRIVER_AUTHOR);
MODULE_DESCRIPTION(DRIVER_DESC);
```

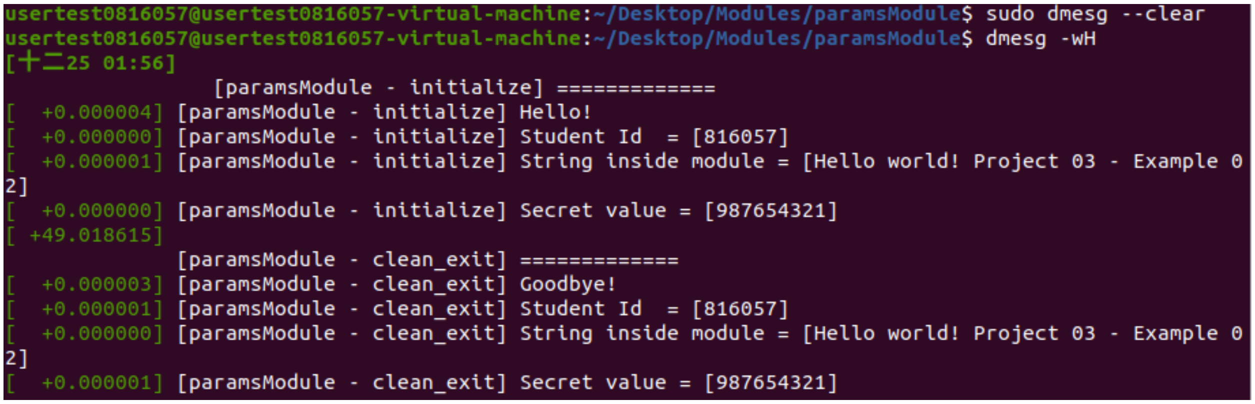


```
obj-m = paramsModule.o

KVERSION = $(shell uname -r)

all:
	make -C /lib/modules/$(KVERSION)/build M=$(PWD) modules

clean:
	make -C /lib/modules/$(KVERSION)/build M=$(PWD) clean
```

Screenshot #8

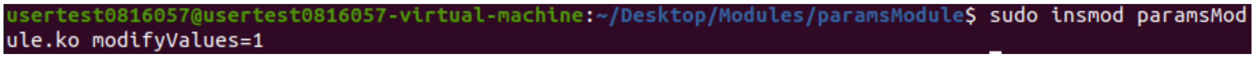


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ make clean
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/paramsModule clean
make[1]: Entering directory '/usr/src/linux-5.13.19'
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ make
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/paramsModule modules
make[1]: Entering directory '/usr/src/linux-5.13.19'
  CC [M]  /home/usertest0816057/Desktop/Modules/paramsModule/paramsModule.o
  MODPOST /home/usertest0816057/Desktop/Modules/paramsModule/Module.symvers
  CC [M]  /home/usertest0816057/Desktop/Modules/paramsModule/paramsModule.mod.o
  LD [M]  /home/usertest0816057/Desktop/Modules/paramsModule/paramsModule.ko
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo insmod paramsMod
ule.ko
[sudo] password for usertest0816057:
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo rmmod paramsModu
le
```

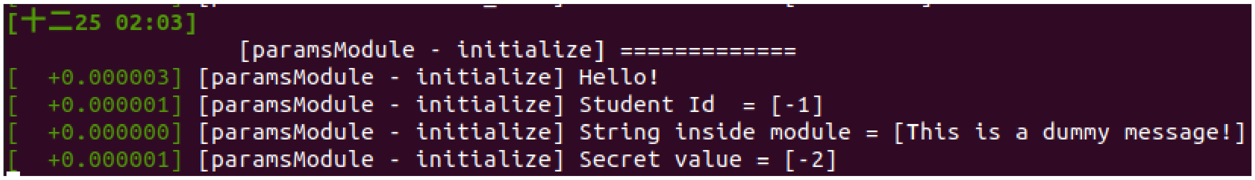


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo dmesg --clear
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ dmesg -wH
[十二25 01:56]
                 [paramsModule - initialize] =============
[  +0.000004] [paramsModule - initialize] Hello!
[  +0.000000] [paramsModule - initialize] Student Id  = [816057]
[  +0.000001] [paramsModule - initialize] String inside module = [Hello world! Project 03 - Example 0
2]
[  +0.000000] [paramsModule - initialize] Secret value = [987654321]
[ +49.018615]
              [paramsModule - clean_exit] =============
[  +0.000003] [paramsModule - clean_exit] Goodbye!
[  +0.000001] [paramsModule - clean_exit] Student Id  = [816057]
[  +0.000000] [paramsModule - clean_exit] String inside module = [Hello world! Project 03 - Example 0
2]
[  +0.000001] [paramsModule - clean_exit] Secret value = [987654321]
```

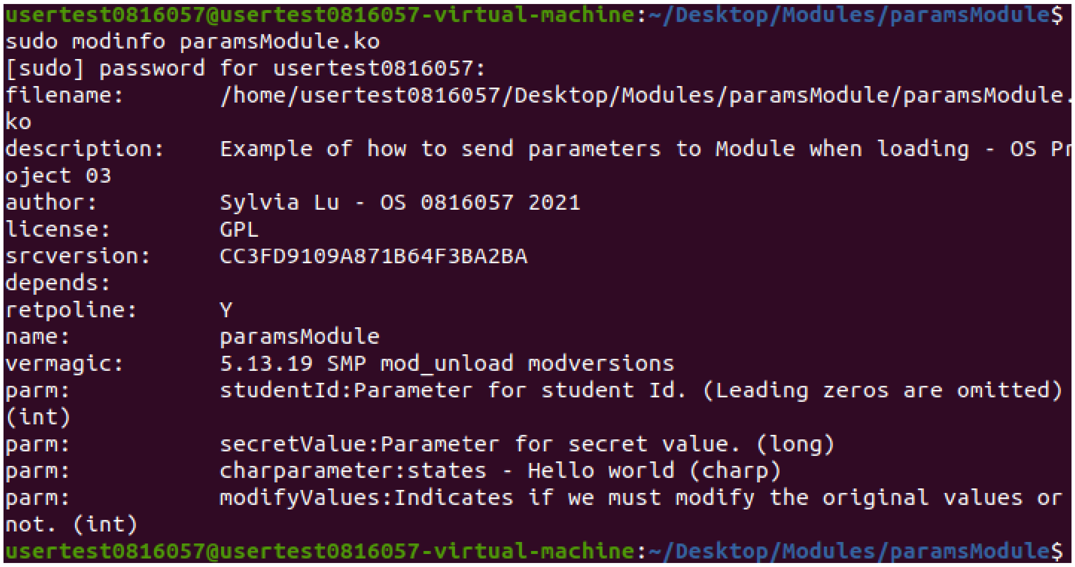
Screenshot #9



```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo insmod paramsMod
ule.ko modifyValues=1
```

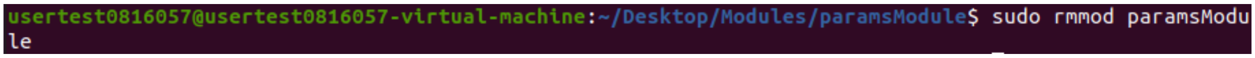


```
[十二25 02:03]
                 [paramsModule - initialize] =============
[  +0.000003] [paramsModule - initialize] Hello!
[  +0.000001] [paramsModule - initialize] Student Id  = [-1]
[  +0.000000] [paramsModule - initialize] String inside module = [This is a dummy message!]
[  +0.000001] [paramsModule - initialize] Secret value = [-2]
```

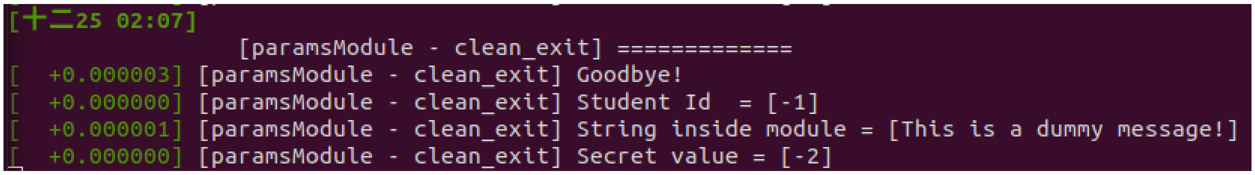


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$
sudo modinfo paramsModule.ko
[sudo] password for usertest0816057:
filename:       /home/usertest0816057/Desktop/Modules/paramsModule/paramsModule.
ko
description:    Example of how to send parameters to Module when loading - OS Pr
oject 03
author:         Sylvia Lu - OS 0816057 2021
license:        GPL
srcversion:     CC3FD9109A871B64F3BA2BA
depends:
retpoline:      Y
name:           paramsModule
vermagic:       5.13.19 SMP mod_unload modversions
parm:           studentId:Parameter for student Id. (Leading zeros are omitted)
(int)
parm:           secretValue:Parameter for secret value. (long)
parm:           charparameter:states - Hello world (charp)
parm:           modifyValues:Indicates if we must modify the original values or
not. (int)
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$
```

Screenshot #10

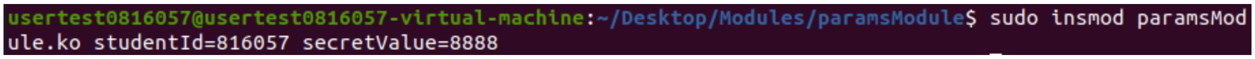


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo rmmod paramsModu
le
```

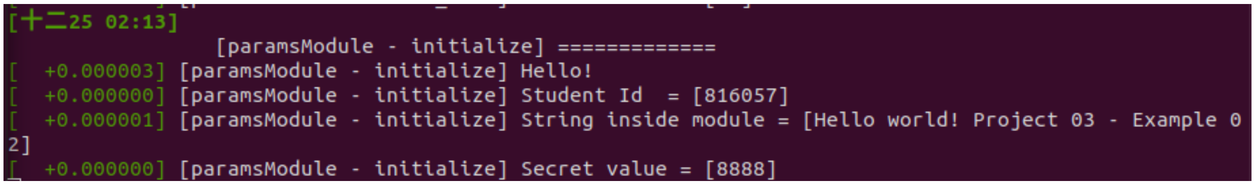


```
[十二25 02:07]
                 [paramsModule - clean_exit] =============
[  +0.000003] [paramsModule - clean_exit] Goodbye!
[  +0.000000] [paramsModule - clean_exit] Student Id  = [-1]
[  +0.000001] [paramsModule - clean_exit] String inside module = [This is a dummy message!]
[  +0.000000] [paramsModule - clean_exit] Secret value = [-2]
```

Screenshot #11

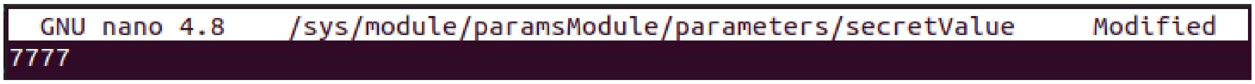


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo insmod paramsMod
ule.ko studentId=816057 secretValue=8888
```



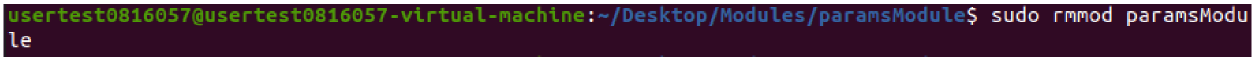
```
[十二25 02:13]
                 [paramsModule - initialize] =============
[  +0.000003] [paramsModule - initialize] Hello!
[  +0.000000] [paramsModule - initialize] Student Id  = [816057]
[  +0.000001] [paramsModule - initialize] String inside module = [Hello world! Project 03 - Example 0
2]
[  +0.000000] [paramsModule - initialize] Secret value = [8888]
```

Screenshot #12

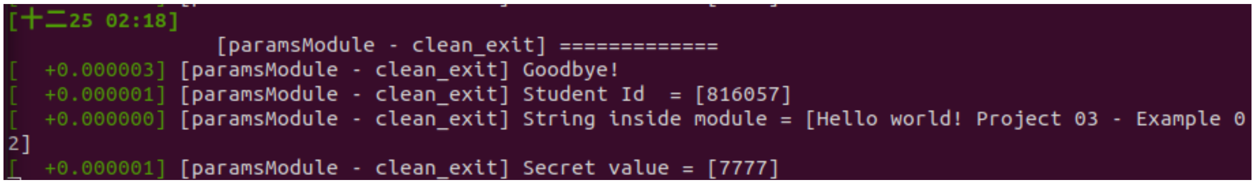


```
  GNU nano 4.8     /sys/module/paramsModule/parameters/secretValue      Modified
7777
```

Screenshot #13

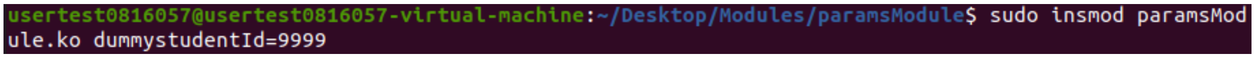


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo rmmod paramsModu
le
```

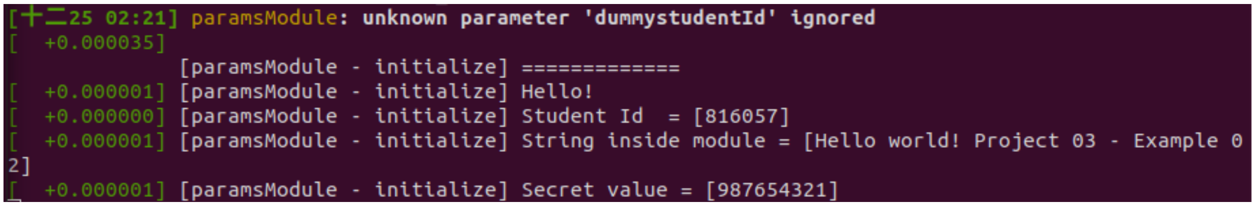


```
[十二25 02:18]
                    [paramsModule - clean_exit] =============
[  +0.000003] [paramsModule - clean_exit] Goodbye!
[  +0.000001] [paramsModule - clean_exit] Student Id  = [816057]
[  +0.000000] [paramsModule - clean_exit] String inside module = [Hello world! Project 03 - Example 0
2]
[  +0.000001] [paramsModule - clean_exit] Secret value = [7777]
```

Screenshot #14

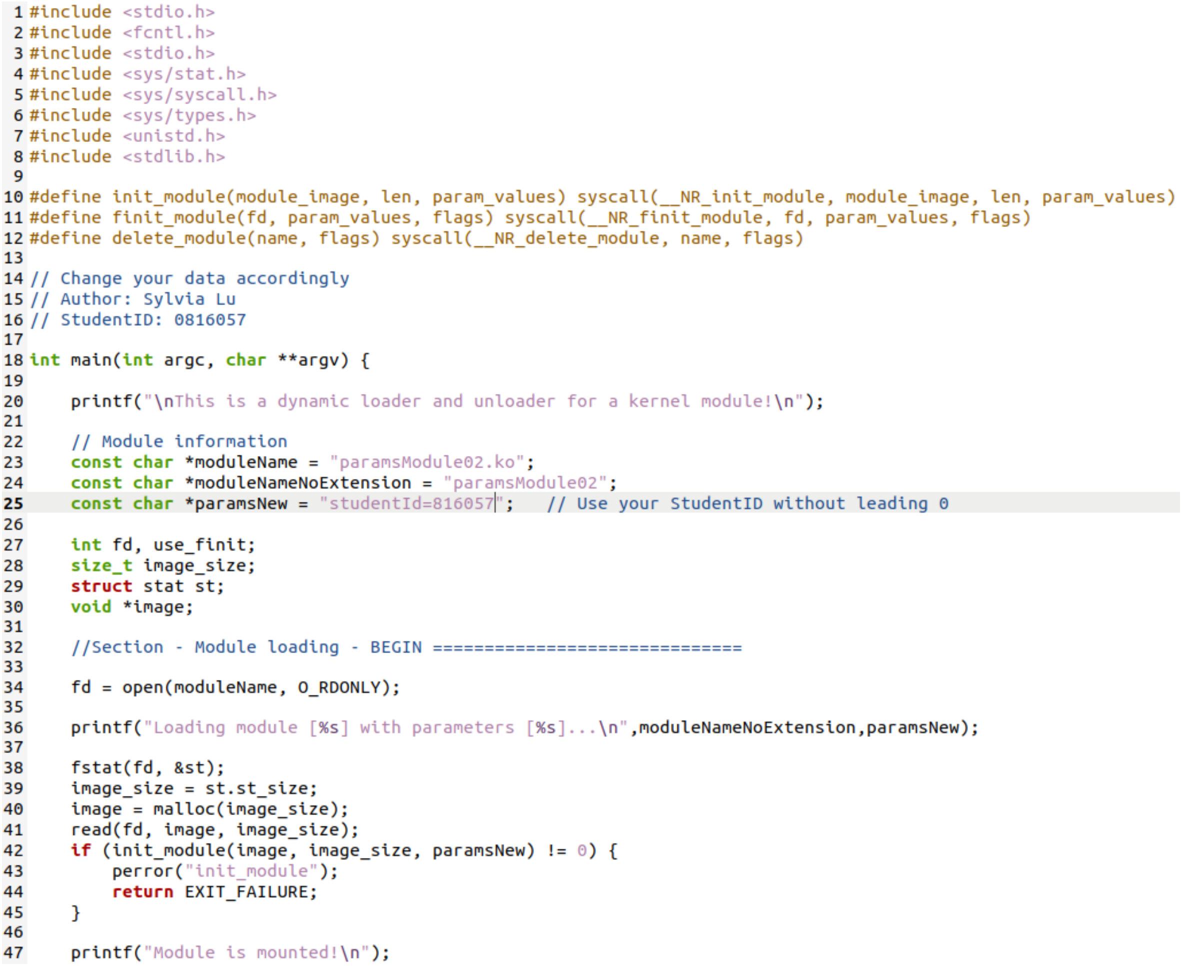


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/paramsModule$ sudo insmod paramsMod
ule.ko dummystudentId=9999
```



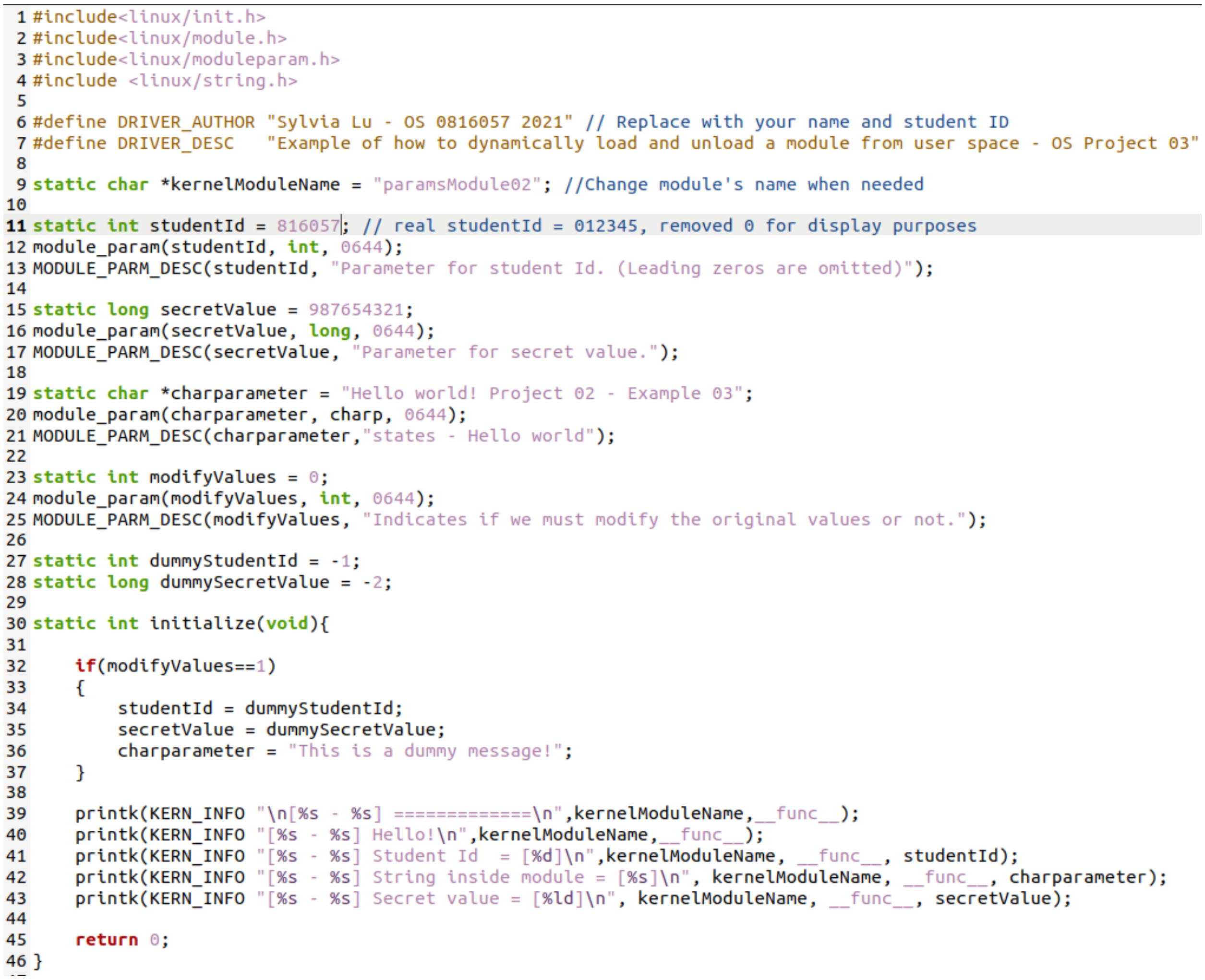
```
[十二25 02:21] paramsModule: unknown parameter 'dummystudentId' ignored
[  +0.000035]
              [paramsModule - initialize] =============
[  +0.000001] [paramsModule - initialize] Hello!
[  +0.000000] [paramsModule - initialize] Student Id  = [816057]
[  +0.000001] [paramsModule - initialize] String inside module = [Hello world! Project 03 - Example 0
2]
[  +0.000001] [paramsModule - initialize] Secret value = [987654321]
```

Screenshot #15

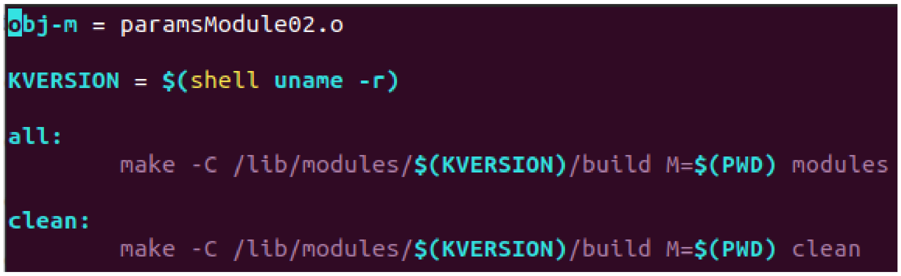


```c
#include <stdio.h>
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/syscall.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>

#define init_module(module_image, len, param_values) syscall(__NR_init_module, module_image, len, param_values)
#define finit_module(fd, param_values, flags) syscall(__NR_finit_module, fd, param_values, flags)
#define delete_module(name, flags) syscall(__NR_delete_module, name, flags)

// Change your data accordingly
// Author: Sylvia Lu
// StudentID: 0816057

int main(int argc, char **argv) {

    printf("\nThis is a dynamic loader and unloader for a kernel module!\n");

    // Module information
    const char *moduleName = "paramsModule02.ko";
    const char *moduleNameNoExtension = "paramsModule02";
    const char *paramsNew = "studentId=816057";    // Use your StudentID without leading 0

    int fd, use_finit;
    size_t image_size;
    struct stat st;
    void *image;

    //Section - Module loading - BEGIN ==============================

    fd = open(moduleName, O_RDONLY);

    printf("Loading module [%s] with parameters [%s]...\n",moduleNameNoExtension,paramsNew);

    fstat(fd, &st);
    image_size = st.st_size;
    image = malloc(image_size);
    read(fd, image, image_size);
    if (init_module(image, image_size, paramsNew) != 0) {
        perror("init_module");
        return EXIT_FAILURE;
    }

    printf("Module is mounted!\n");
```

Screenshot #16

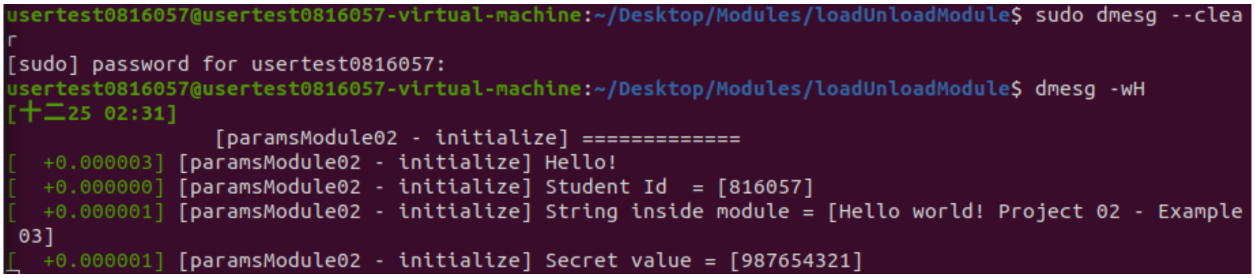


```c
#include<linux/init.h>
#include<linux/module.h>
#include<linux/moduleparam.h>
#include <linux/string.h>

#define DRIVER_AUTHOR "Sylvia Lu - OS 0816057 2021" // Replace with your name and student ID
#define DRIVER_DESC   "Example of how to dynamically load and unload a module from user space - OS Project 03"

static char *kernelModuleName = "paramsModule02"; //Change module's name when needed

static int studentId = 816057; // real studentId = 012345, removed 0 for display purposes
module_param(studentId, int, 0644);
MODULE_PARM_DESC(studentId, "Parameter for student Id. (Leading zeros are omitted)");

static long secretValue = 987654321;
module_param(secretValue, long, 0644);
MODULE_PARM_DESC(secretValue, "Parameter for secret value.");

static char *charparameter = "Hello world! Project 02 - Example 03";
module_param(charparameter, charp, 0644);
MODULE_PARM_DESC(charparameter,"states - Hello world");

static int modifyValues = 0;
module_param(modifyValues, int, 0644);
MODULE_PARM_DESC(modifyValues, "Indicates if we must modify the original values or not.");

static int dummyStudentId = -1;
static long dummySecretValue = -2;

static int initialize(void){

    if(modifyValues==1)
    {
        studentId = dummyStudentId;
        secretValue = dummySecretValue;
        charparameter = "This is a dummy message!";
    }

    printk(KERN_INFO "\n[%s - %s] =============\n",kernelModuleName,__func__);
    printk(KERN_INFO "[%s - %s] Hello!\n",kernelModuleName,__func__);
    printk(KERN_INFO "[%s - %s] Student Id  = [%d]\n",kernelModuleName, __func__, studentId);
    printk(KERN_INFO "[%s - %s] String inside module = [%s]\n", kernelModuleName, __func__, charparameter);
    printk(KERN_INFO "[%s - %s] Secret value = [%ld]\n", kernelModuleName, __func__, secretValue);

    return 0;
}
```

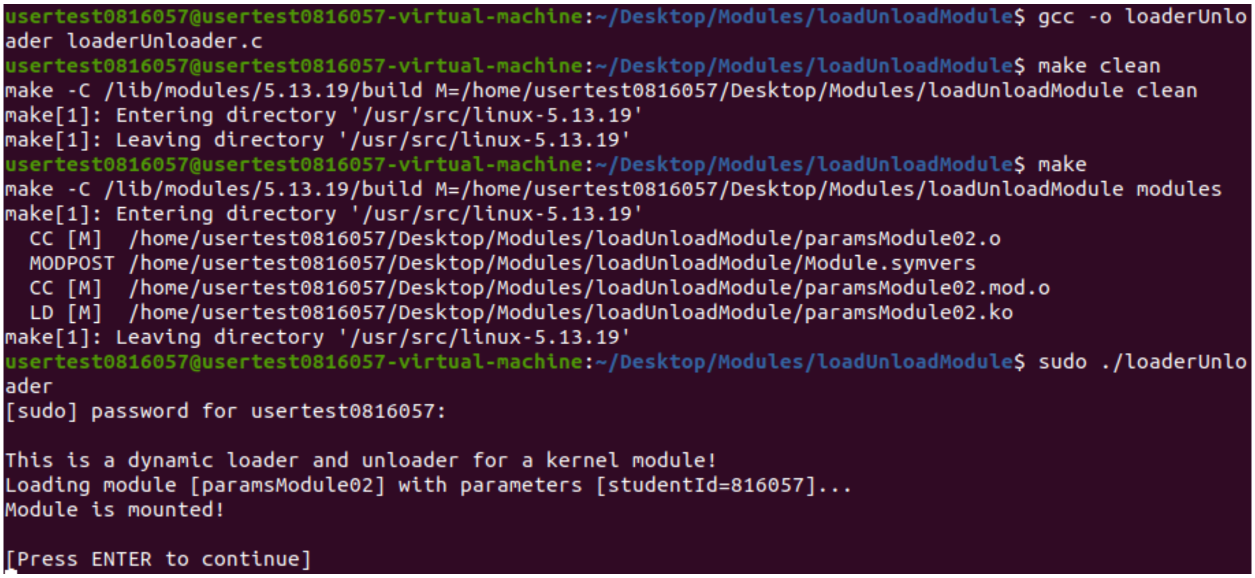


```makefile
obj-m = paramsModule02.o

KVERSION = $(shell uname -r)

all:
	make -C /lib/modules/$(KVERSION)/build M=$(PWD) modules

clean:
	make -C /lib/modules/$(KVERSION)/build M=$(PWD) clean
```

Screenshot #17

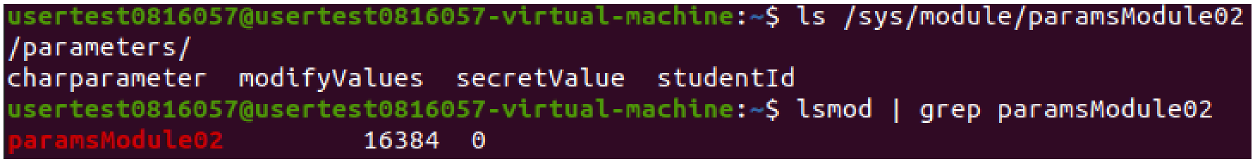


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$ sudo dmesg --clea
r
[sudo] password for usertest0816057:
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$ dmesg -wH
[十二25 02:31]
                [paramsModule02 - initialize] =============
[  +0.000003] [paramsModule02 - initialize] Hello!
[  +0.000000] [paramsModule02 - initialize] Student Id  = [816057]
[  +0.000001] [paramsModule02 - initialize] String inside module = [Hello world! Project 02 - Example
 03]
[  +0.000001] [paramsModule02 - initialize] Secret value = [987654321]
```

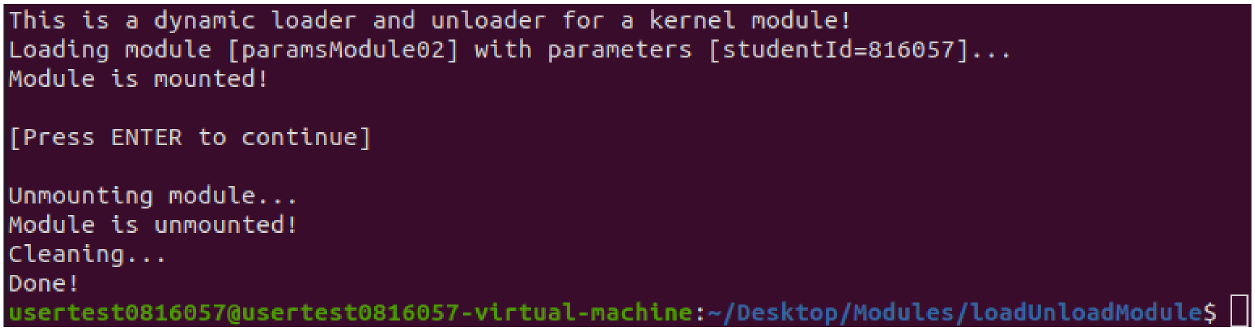


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$ gcc -o loaderUnlo
ader loaderUnloader.c
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$ make clean
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/loadUnloadModule clean
make[1]: Entering directory '/usr/src/linux-5.13.19'
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$ make
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/loadUnloadModule modules
make[1]: Entering directory '/usr/src/linux-5.13.19'
  CC [M]  /home/usertest0816057/Desktop/Modules/loadUnloadModule/paramsModule02.o
  MODPOST /home/usertest0816057/Desktop/Modules/loadUnloadModule/Module.symvers
  CC [M]  /home/usertest0816057/Desktop/Modules/loadUnloadModule/paramsModule02.mod.o
  LD [M]  /home/usertest0816057/Desktop/Modules/loadUnloadModule/paramsModule02.ko
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$ sudo ./loaderUnlo
ader
[sudo] password for usertest0816057:

This is a dynamic loader and unloader for a kernel module!
Loading module [paramsModule02] with parameters [studentId=816057]...
Module is mounted!

[Press ENTER to continue]
```

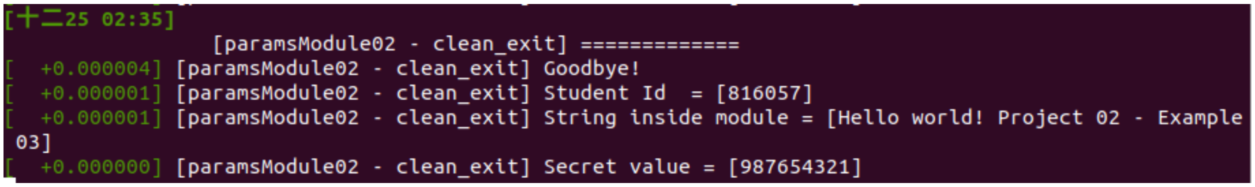
Screenshot #18



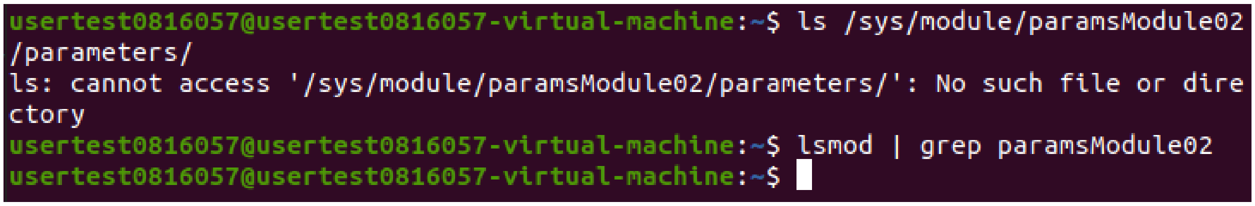
```
usertest0816057@usertest0816057-virtual-machine:~$ ls /sys/module/paramsModule02
/parameters/
charparameter  modifyValues  secretValue  studentId
usertest0816057@usertest0816057-virtual-machine:~$ lsmod | grep paramsModule02
paramsModule02         16384  0
```



```
This is a dynamic loader and unloader for a kernel module!
Loading module [paramsModule02] with parameters [studentId=816057]...
Module is mounted!

[Press ENTER to continue]

Unmounting module...
Module is unmounted!
Cleaning...
Done!
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/loadUnloadModule$
```

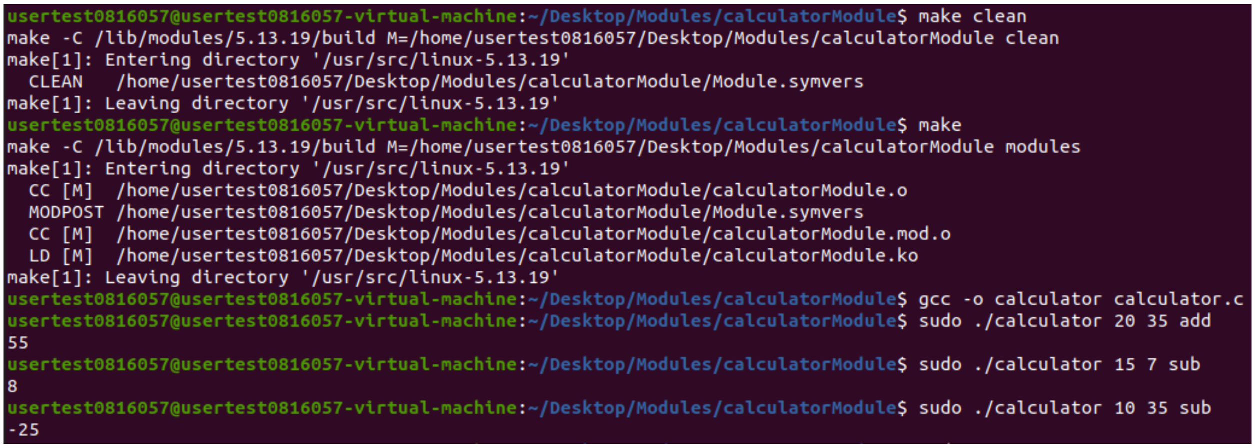


```
[十二25 02:35]
                 [paramsModule02 - clean_exit] =============
[  +0.000004] [paramsModule02 - clean_exit] Goodbye!
[  +0.000001] [paramsModule02 - clean_exit] Student Id  = [816057]
[  +0.000001] [paramsModule02 - clean_exit] String inside module = [Hello world! Project 02 - Example
 03]
[  +0.000000] [paramsModule02 - clean_exit] Secret value = [987654321]
```

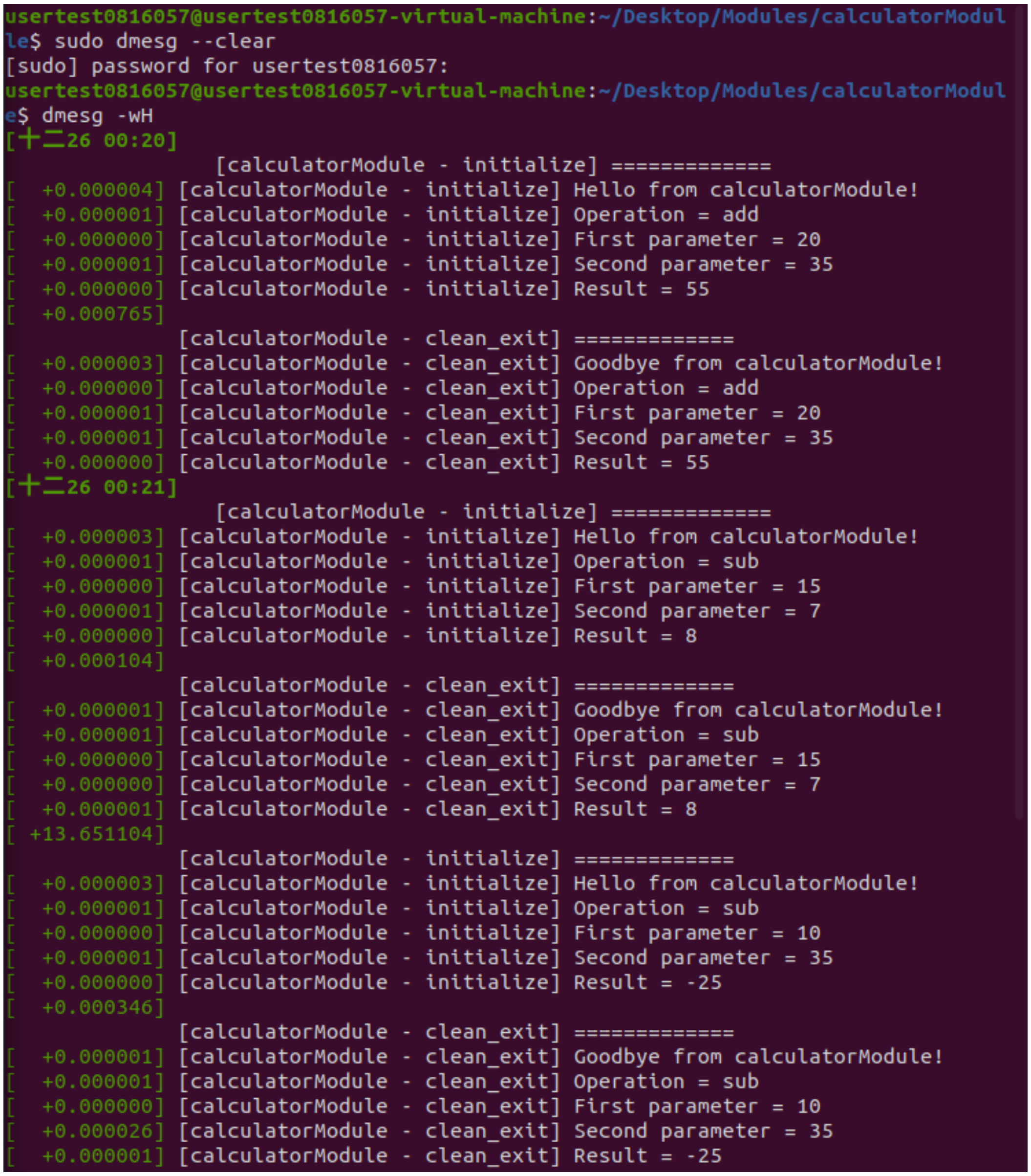


```
usertest0816057@usertest0816057-virtual-machine:~$ ls /sys/module/paramsModule02
/parameters/
ls: cannot access '/sys/module/paramsModule02/parameters/': No such file or dire
ctory
usertest0816057@usertest0816057-virtual-machine:~$ lsmod | grep paramsModule02
usertest0816057@usertest0816057-virtual-machine:~$
```

Screenshot #19

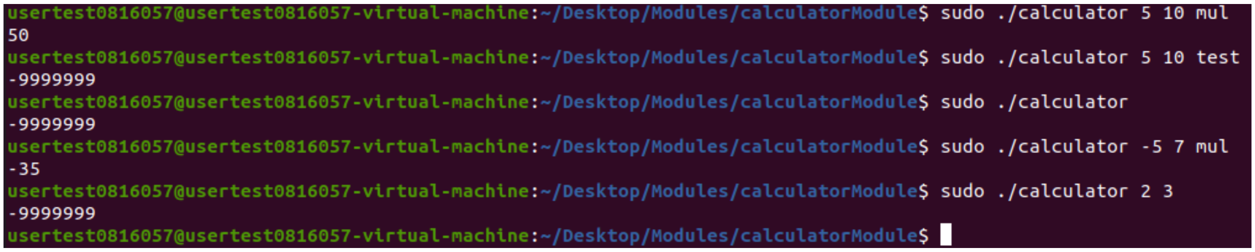


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ make clean
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/calculatorModule clean
make[1]: Entering directory '/usr/src/linux-5.13.19'
  CLEAN   /home/usertest0816057/Desktop/Modules/calculatorModule/Module.symvers
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ make
make -C /lib/modules/5.13.19/build M=/home/usertest0816057/Desktop/Modules/calculatorModule modules
make[1]: Entering directory '/usr/src/linux-5.13.19'
  CC [M]  /home/usertest0816057/Desktop/Modules/calculatorModule/calculatorModule.o
  MODPOST /home/usertest0816057/Desktop/Modules/calculatorModule/Module.symvers
  CC [M]  /home/usertest0816057/Desktop/Modules/calculatorModule/calculatorModule.mod.o
  LD [M]  /home/usertest0816057/Desktop/Modules/calculatorModule/calculatorModule.ko
make[1]: Leaving directory '/usr/src/linux-5.13.19'
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ gcc -o calculator calculator.c
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator 20 35 add
55
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator 15 7 sub
8
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator 10 35 sub
-25
```

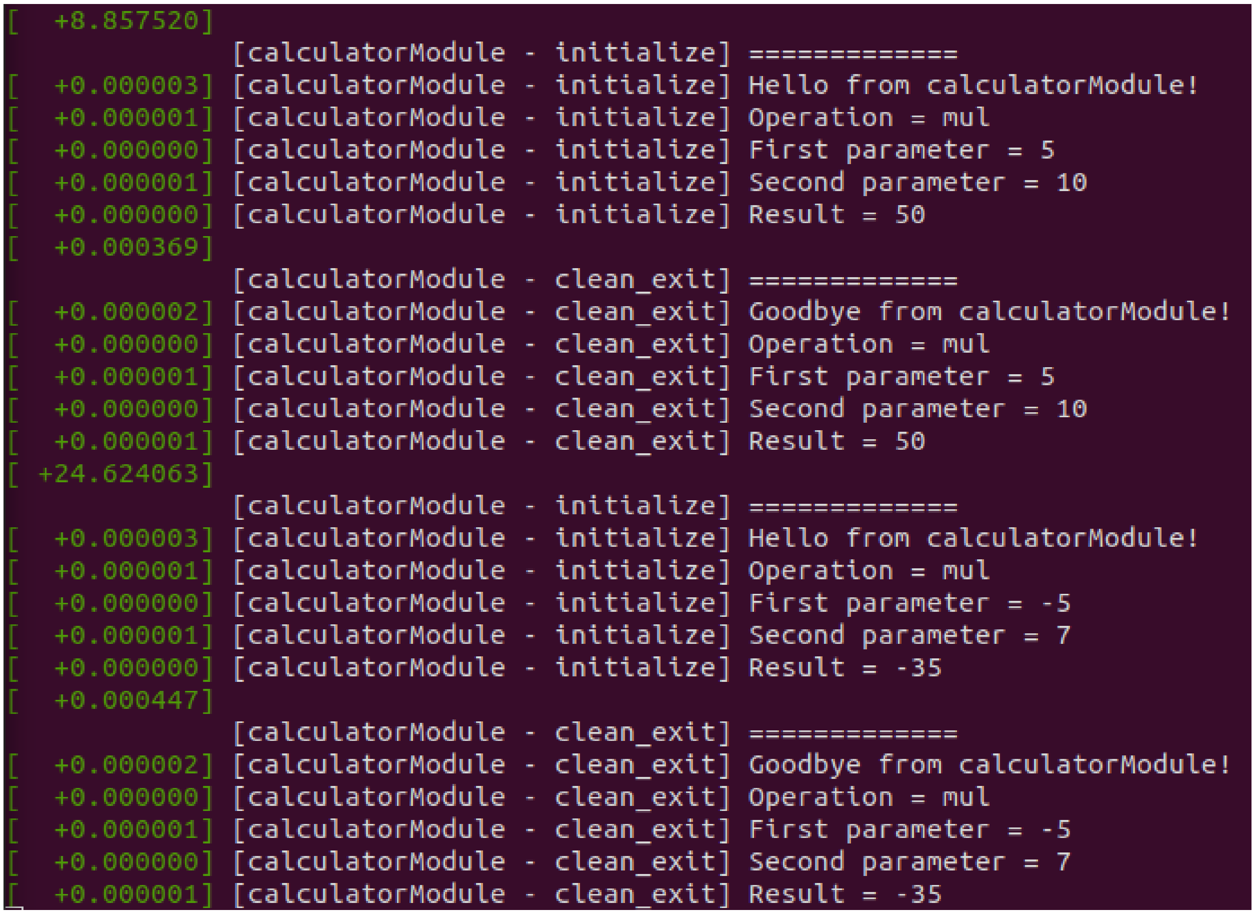


```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModul
le$ sudo dmesg --clear
[sudo] password for usertest0816057:
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModul
e$ dmesg -wH
[十二26 00:20]
                 [calculatorModule - initialize] =============
[  +0.000004] [calculatorModule - initialize] Hello from calculatorModule!
[  +0.000001] [calculatorModule - initialize] Operation = add
[  +0.000000] [calculatorModule - initialize] First parameter = 20
[  +0.000001] [calculatorModule - initialize] Second parameter = 35
[  +0.000000] [calculatorModule - initialize] Result = 55
[  +0.000765]
              [calculatorModule - clean_exit] =============
[  +0.000003] [calculatorModule - clean_exit] Goodbye from calculatorModule!
[  +0.000000] [calculatorModule - clean_exit] Operation = add
[  +0.000001] [calculatorModule - clean_exit] First parameter = 20
[  +0.000001] [calculatorModule - clean_exit] Second parameter = 35
[  +0.000000] [calculatorModule - clean_exit] Result = 55
[十二26 00:21]
                 [calculatorModule - initialize] =============
[  +0.000003] [calculatorModule - initialize] Hello from calculatorModule!
[  +0.000001] [calculatorModule - initialize] Operation = sub
[  +0.000000] [calculatorModule - initialize] First parameter = 15
[  +0.000001] [calculatorModule - initialize] Second parameter = 7
[  +0.000000] [calculatorModule - initialize] Result = 8
[  +0.000104]
              [calculatorModule - clean_exit] =============
[  +0.000001] [calculatorModule - clean_exit] Goodbye from calculatorModule!
[  +0.000001] [calculatorModule - clean_exit] Operation = sub
[  +0.000000] [calculatorModule - clean_exit] First parameter = 15
[  +0.000000] [calculatorModule - clean_exit] Second parameter = 7
[  +0.000001] [calculatorModule - clean_exit] Result = 8
[ +13.651104]
              [calculatorModule - initialize] =============
[  +0.000003] [calculatorModule - initialize] Hello from calculatorModule!
[  +0.000001] [calculatorModule - initialize] Operation = sub
[  +0.000000] [calculatorModule - initialize] First parameter = 10
[  +0.000001] [calculatorModule - initialize] Second parameter = 35
[  +0.000000] [calculatorModule - initialize] Result = -25
[  +0.000346]
              [calculatorModule - clean_exit] =============
[  +0.000001] [calculatorModule - clean_exit] Goodbye from calculatorModule!
[  +0.000001] [calculatorModule - clean_exit] Operation = sub
[  +0.000000] [calculatorModule - clean_exit] First parameter = 10
[  +0.000026] [calculatorModule - clean_exit] Second parameter = 35
[  +0.000001] [calculatorModule - clean_exit] Result = -25
```

Screenshot #20



```
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator 5 10 mul
50
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator 5 10 test
-9999999
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator
-9999999
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator -5 7 mul
-35
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ sudo ./calculator 2 3
-9999999
usertest0816057@usertest0816057-virtual-machine:~/Desktop/Modules/calculatorModule$ 
```



```
[   +8.857520]
               [calculatorModule - initialize] =============
[   +0.000003] [calculatorModule - initialize] Hello from calculatorModule!
[   +0.000001] [calculatorModule - initialize] Operation = mul
[   +0.000000] [calculatorModule - initialize] First parameter = 5
[   +0.000001] [calculatorModule - initialize] Second parameter = 10
[   +0.000000] [calculatorModule - initialize] Result = 50
[   +0.000369]
               [calculatorModule - clean_exit] =============
[   +0.000002] [calculatorModule - clean_exit] Goodbye from calculatorModule!
[   +0.000000] [calculatorModule - clean_exit] Operation = mul
[   +0.000001] [calculatorModule - clean_exit] First parameter = 5
[   +0.000000] [calculatorModule - clean_exit] Second parameter = 10
[   +0.000001] [calculatorModule - clean_exit] Result = 50
[  +24.624063]
               [calculatorModule - initialize] =============
[   +0.000003] [calculatorModule - initialize] Hello from calculatorModule!
[   +0.000001] [calculatorModule - initialize] Operation = mul
[   +0.000000] [calculatorModule - initialize] First parameter = -5
[   +0.000001] [calculatorModule - initialize] Second parameter = 7
[   +0.000000] [calculatorModule - initialize] Result = -35
[   +0.000447]
               [calculatorModule - clean_exit] =============
[   +0.000002] [calculatorModule - clean_exit] Goodbye from calculatorModule!
[   +0.000000] [calculatorModule - clean_exit] Operation = mul
[   +0.000001] [calculatorModule - clean_exit] First parameter = -5
[   +0.000000] [calculatorModule - clean_exit] Second parameter = 7
[   +0.000001] [calculatorModule - clean_exit] Result = -35
```