New York University
Tandon School of Engineering
Department of Computer Science and Engineering

Introduction to Operating Systems
Fall 2025

Assignment 3
(10 points)

Develop a simple Linux kernel module that runs on your virtual machine. The only functionality required of your module is to be able to load and unload, printing a debug message while doing so.

When a Linux kernel module is loaded, it invokes an init function, and when it is removed (or unloaded), it invokes an exit function.

A) (0 points) Read chapter 2 of the freely available O'Reilly book "Linux Device Drivers, 3rd Edition" (https://lwn.net/Kernel/LDD3/), in particular p.16, as well as your text book p.96 to get you started. Note that even though the LDD3 book is written for kernel version 2.6, most mechanisms are applicable with minor or no changes. The relevant example code is copied below as a starting point.

```
#include <linux/init.h>
#include <linux/module.h>
MODULE_LICENSE("Dual BSD/GPL");
static int hello_init(void)
{
      printk(KERN_ALERT "Hello, world\n");
      return 0;
}
static void hello_exit(void)
{
      printk(KERN_ALERT "Goodbye, cruel world\n");
}
module_init(hello_init);
module_exit(hello_exit);
```

The `hello_init()` function is invoked when you insert your module (using the `insmod` shell command), whereas the `hello_exit()` is called when you unload your module (using the `rmmod` shell command).

B) (0 points) Read the description of the global kernel variable `jiffies` and the macro `HZ` in the O'Reilly book (search in the searchable pdf). Then read about the `ktime_get_boottime`() routine in https://www.kernel.org/doc/html/latest/core-api/timekeeping.html

C) (0 points) You may need to install the kernel headers if not already installed. Type:

```
sudo apt-get install linux-headers-$(uname -r)
```

D) (10 points) Modify the c code given above (which becomes your lab3.c) such that:
1) The init function prints a "hello" string, followed by the tick time in milliseconds (i.e. the timer interval, as we defined it in weeks 1/2).

It then prints the current time on the next line, preceded by the string "Curent time is: ". You shall format the current time in **hh:mm:ss**, i.e. hours, minutes and seconds. Don't worry about converting from the GMT standard time into your time zone, do not worry about converting to 12 hour clock and do not print the date.
The init function shall also save the value of `jiffies` and the current time.
Hint: you may use `ktime_get_real()`, `ktime_get_real_ns()` or `ktime_get_real_ts64(struct timespec64*)`. Further info may be found in https://docs.kernel.org/core-api/timekeeping.html#)

2) The exit function prints a "goodbye" string, followed by the time **in milliseconds** between the insertion and removal of the module i.e. between init and exit functions) using the difference in the value of jiffies from inserting the module to removing the module.
On the following line, it prints the current time, in the same manner the init function did.

You shall use the Makefile provided with the assignment (In some cases, you may need to slightly modify the Makefile provided to suit your setup). You should place it in the same directory as your .c file (lab3.c)

Hints:

E) Your module should use `printk()` to print messages. You will use this print facility to also debug your code if needed (😊). More information may be found on https://www.kernel.org/doc/html/latest/core-api/printk-basics.html

F) Use `dmesg` shell command to view messages printed by `printk()`, e.g. type:

```
dmesg
```

You may clear the log using:

```
dmesg -C
```

What to submit to gradescope:

Please submit the following files individually:

1) Source file(s) with appropriate comments.
The naming should be similar to "**lab#_$.c**" (**#** is replaced with the assignment number and **$** with the question number within the assignment, e.g. lab4_b.c, for lab 4, question b OR lab5_1a for lab 5, question 1a).
2) A single pdf file (for images + report/answers to questions), named **"lab#.pdf"** (**#** is replaced by the assignment number), containing:
    - Screen shot(**s**) of your terminal window showing the current directory, the command used to compile your program, the command used to run your program and the output of your program.
3) Your Makefile, if any. This is applicable only to kernel modules.

RULES:

- You shall **use kernel version 4.x.x or above**. You shall not use kernel version 3.x.x.
- You may consult with other students about GENERAL concepts or methods but copying code (or code fragments) or algorithms is NOT ALLOWED and is considered cheating (whether copied form other students, the internet or any other source).
- If you are having trouble, please ask your teaching assistant for help.
- You must submit your assignment prior to the deadline.