Phil Nevins
ECE 373
Assignment 1

A. Information Gathering [<u>Deliverable #1</u>]

1.

a. BUGS when using open():
Currently, it is not possible to enable signal-driven I/O by specifying O_ASYNC when calling open(); use fcntl(2) to enable this flag.

One must check for two different error codes, EISDIR and ENOENT, when trying to determine whether the kernel supports O_TMPFILE functionality.

When both O_CREAT and O_DIRECTORY are specified in flags and the file specified by pathname does not exist, open() will create a regular file (i.e., O_DIRECTORY is ignored).

b. Files needed to use open():
sys/types.h, sys/stat.h, fcntl.h

c. Three system calls associated with open():
creat(), openat(), openat2(2)

d. I chose creat()

BUGS:
Currently, it is not possible to enable signal-driven I/O by specifying O_ASYNC when calling open(); use fcntl(2) to enable this flag.

One must check for two different error codes, EISDIR and ENOENT, when trying to determine whether the kernel supports O_TMPFILE functionality.

When both O_CREAT and O_DIRECTORY are specified in flags and the file specified by pathname does not exist, open() will create a regular file (i.e., O_DIRECTORY is ignored).

Files needed to use create():
sys/types.h, sys/stat.h, fcnt1.h

2.

a. File where the struct is defined: include/linux/usb.h
First five members of the struct: devnum, devpath[16], route, state, speed

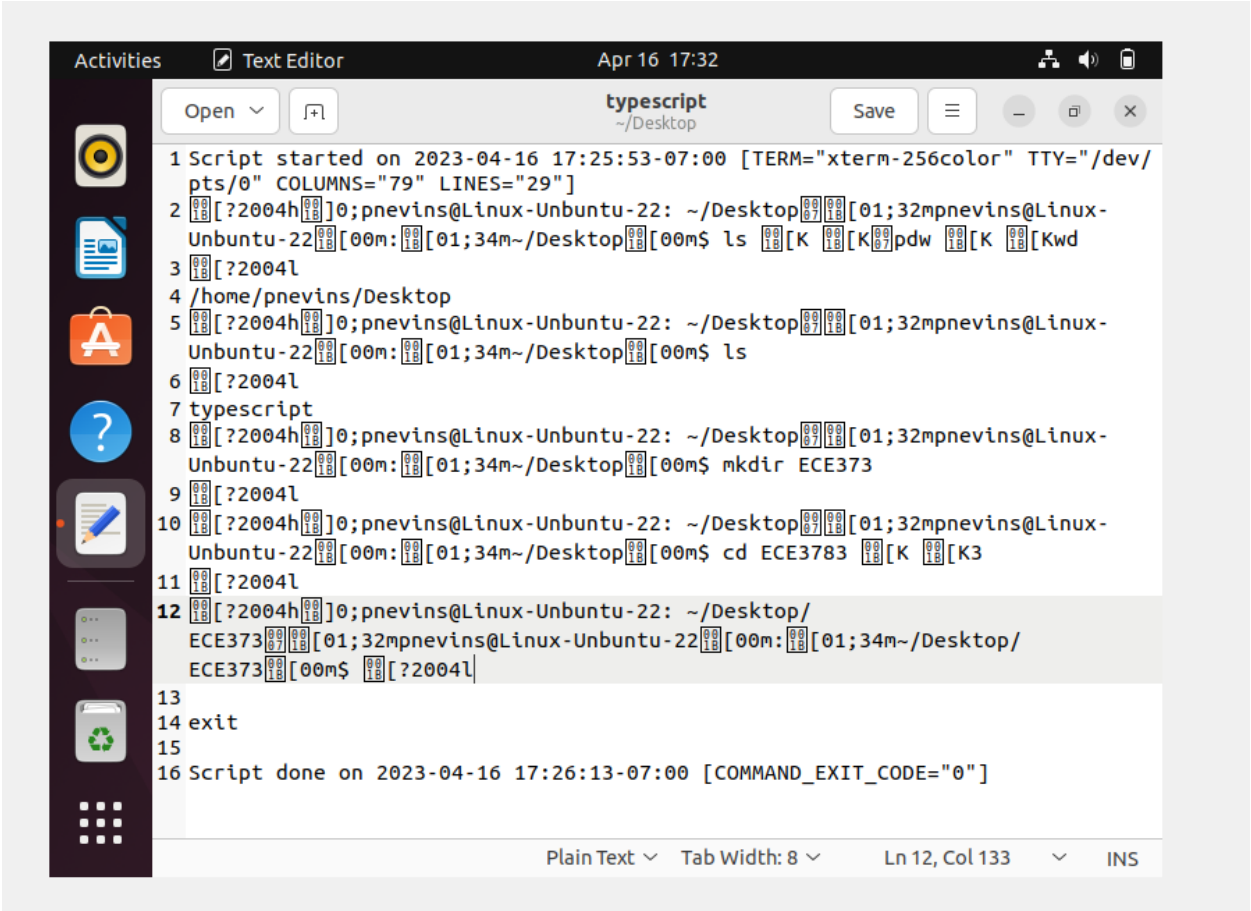
b. Include/uapi/linux/usb/ch9.h

c. **Deliverable #2**

Searched for usb_device_speed and selected the .h file where it is defined as an enum

```
enum usb_device_speed {
        USB_SPEED_UNKNOWN = 0,                  /* enumerating */
        USB_SPEED_LOW, USB_SPEED_FULL,          /* usb 1.1 */
        USB_SPEED_HIGH,                         /* usb 2.0 */
        USB_SPEED_WIRELESS,                     /* wireless (usb 2.5) */
        USB_SPEED_SUPER,                /* usb 3.0 */
        USB_SPEED_SUPER_PLUS,                   /* usb 3.1 */
};
```

B. Basic Linux Use



Typescript from Part B

C. Basic C Programming in Linux [Deliverables #3 and #4]

Deliverable #3

I made a simple Celsius to Fahrenheit conversion program. The hello.c program from the link in the assignment didn't work and was missing a ton of libraries.

```
#include <stdio.h>

int main() {
   float celsius, fahrenheit;

   printf("Enter temperature in Celsius: ");
   scanf("%f", &celsius);

   fahrenheit = (celsius * 9 / 5) + 32;

   printf("%.2f Celsius is %.2f Fahrenheit.\n", celsius, fahrenheit);

   return 0;
}
```

Deliverable #4

```
Script started on 2023-04-16 18:17:37-07:00 [TERM="xterm-256color" TTY="/dev/pts/4" COLUMNS="79" LINES="29"]
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop[01;32mpnevins@Linux-Unbuntu-22[00m:[01;34m~/Desktop[00m$ cd ECE373
[?2004l
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-22[00m:[01;34m~/Desktop/ECE373[00m$ gcc -g -i [K[Ko CtF[KoT[KF CtoF.c
[?2004l
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-22[00m:[01;34m~/Desktop/ECE373[00m$ ./CtoF
[?2004l
Enter temperature in Celsius: 0
0.00 Celsius is 32.00 Fahrenheit.
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-22[00m:[01;34m~/Desktop/ECE373[00m$ [?2004l

exit

Script done on 2023-04-16 18:18:08-07:00 [COMMAND_EXIT_CODE="0"]
```

D. Hello, Kernal

Deliverable #5 (Code taken from slides)

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

MODULE_LICENSE("Dual BSD/GPL");

static int __init hello_init(void)
{
	printk(KERN_INFO "Hello, kernel\n");
	return 0;

}

static void __exit hello_exit(void)
{
	printk(KERN_INFO "Goodbye, kernel\n");
}

module_init(hello_init);
module_exit(hello_exit);
```

Deliverable #6 (logs from transcript and dmesg)

```
Script started on 2023-04-19 14:43:21-07:00 [TERM="xterm-256color" TTY="/dev/pts/3"
COLUMNS="80" LINES="24"]
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-
22[00m:[01;34m~/Desktop/ECE373[00m$ sus[Kdo insmod hello_kernel [K.ko
[?2004l
[sudo] password for pnevins:
insmod: ERROR: could not insert module hello_kernel.ko: File exists
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-
22[00m:[01;34m~/Desktop/ECE373[00m$ sudo rmmod hello.[K_kernel.ko
[?2004l
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-
22[00m:[01;34m~/Desktop/ECE373[00m$ lsmod | grep hello
[?2004l
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-
22[00m:[01;34m~/Desktop/ECE373[00m$ sudio=[K[K[Koi [K[K insmod
hello_+[Kkerna[Kel.ko
[?2004l
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-
22[00m:[01;34m~/Desktop/ECE373[00m$ sudo dsme[K[Kmeg
```

[?2004l
sudo: dsmeg: command not found

[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-22[00m:[01;34m~/Desktop/ECE373[00m$ sudo sm[K[Kdmesg
[?2004l

[32m[1964.529515] Hello, kernel
[32m[2804.871497] Goodbye, kernel
[?2004h]0;pnevins@Linux-Unbuntu-22: ~/Desktop/ECE373[01;32mpnevins@Linux-Unbuntu-22[00m:[01;34m~/Desktop/ECE373[00m$ exit
[?2004l
exit

Script done on 2023-04-19 14:46:01-07:00 [COMMAND_EXIT_CODE="0"]