Phil Nevins  
ECE 373  
Assignment 1

**A. Information Gathering [Deliverable #1]**

2. 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).

1. Files needed to use open():

sys/types.h, sys/stat.h, fcntl.h

1. Three system calls associated with open():

creat(), openat(), openat2(2)

1. 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. File where the struct is defined: include/linux/usb.h

First five members of the struct: devnum, devpath[16], route, state, speed

1. Include/uapi/linux/usb/ch9.h
2. **Deliverable #2**

Searched for usb\_device\_speed and selected the .h file where it is defined as an enum

enum [**usb\_device\_speed**](https://elixir.bootlin.com/linux/latest/C/ident/usb_device_speed) {

[**USB\_SPEED\_UNKNOWN**](https://elixir.bootlin.com/linux/latest/C/ident/USB_SPEED_UNKNOWN) = 0, */\* enumerating \*/*

[**USB\_SPEED\_LOW**](https://elixir.bootlin.com/linux/latest/C/ident/USB_SPEED_LOW), [**USB\_SPEED\_FULL**](https://elixir.bootlin.com/linux/latest/C/ident/USB_SPEED_FULL), */\* usb 1.1 \*/*

[**USB\_SPEED\_HIGH**](https://elixir.bootlin.com/linux/latest/C/ident/USB_SPEED_HIGH), */\* usb 2.0 \*/*

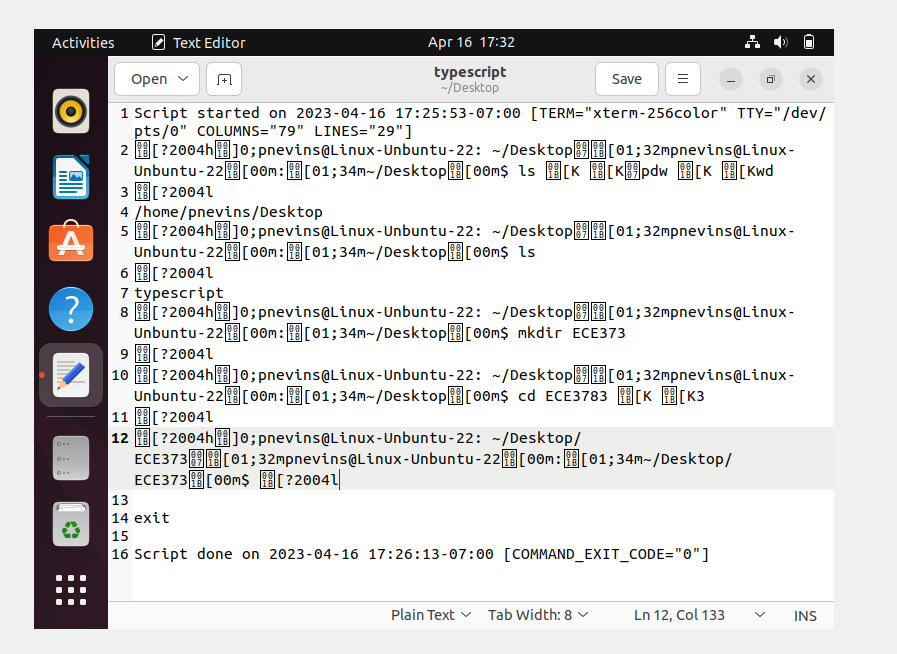
[**USB\_SPEED\_WIRELESS**](https://elixir.bootlin.com/linux/latest/C/ident/USB_SPEED_WIRELESS), */\* wireless (usb 2.5) \*/*

[**USB\_SPEED\_SUPER**](https://elixir.bootlin.com/linux/latest/C/ident/USB_SPEED_SUPER), */\* usb 3.0 \*/*

[**USB\_SPEED\_SUPER\_PLUS**](https://elixir.bootlin.com/linux/latest/C/ident/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"]