

ECE 373 Assignment #1
Spring 2021

A. Information Gathering

1. A. 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 (O_DIRECTORY is ignored).

B. Need to include the following header files: <sys/types.h>, <sys/stat.h>, and <fcntl.h>

C. The first three related system calls are: 'chmod()', 'chown()', and 'close()'

D. I have chosen 'close()'.

There doesn't seem to be any bugs in 'close()', but there are errors.

There is only one header file needed to use system call: <unistd.h>

2. A. Using kernel version: v5.11.13. The "usb_device" structure is defined in <usb.h> and the first five members of the struct are: 'devnum', 'devpath', 'route', 'state', 'speed'.

B. The type for the fifth member is 'usb_device_state' which is defined in the <ch9.h> header file.

C. The full enum declaration for the fifth struct member is:

```
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 */
};
```

ECE 373 Assignment #1
Spring 2021

B. Basic Linux Use

```
Script started on 2021-04-11 13:58:06-07:00 [TERM="xterm-256color" TTY="/dev/pts/0" COLUMNS="79" LINES="24"]
^[]0;ryan@ryan-ece373: ~^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~^[]00m$ pwd
^M
/home/ryan^M
^[]0;ryan@ryan-ece373: ~^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~^[]00m$ ls
-l^M
total 36^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]0m:^[]01;34mDesktop^[]0m^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mDocuments^[]0m^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mDownloads^[]0m^M
drwxrwxr-x 3 ryan ryan 4096 Apr 11 11:28 ^[]01;34mgithub^[]0m^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mMusic^[]0m^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mPictures^[]0m^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mPublic^[]0m^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mTemplates^[]0m^M
-rw-rw-r-- 1 ryan ryan 0 Apr 11 13:58 typescript^M
drwxr-xr-x 2 ryan ryan 4096 Apr 10 23:59 ^[]01;34mVideos^[]0m^M
^[]0;ryan@ryan-ece373: ~^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~^[]00m$ mkdir
ECE373^M
^[]0;ryan@ryan-ece373: ~^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~^[]00m$ cd
ECE373/^M
^[]0;ryan@ryan-ece373: ~/ECE373^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~/ECE
373^[]00m$ exit^M

Script done on 2021-04-11 13:59:51-07:00 [COMMAND_EXIT_CODE="0"]
```

C. Basic C Programming in Linux

Typescript:

```
Script started on 2021-04-14 08:33:20-07:00 [TERM="xterm-256color" TTY="/dev/pts/0" COLUMNS="79" LINES="31"]
^[]0;ryan@ryan-ece373: ~/ECE373^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~/ECE
373^[]00m$ gcc -g -o simpleC simpleC.c^M
^[]0;ryan@ryan-ece373: ~/ECE373^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~/ECE
373^[]00m$ ./simple^H^[]KpleC^M
Enter temperature in Fahrenheit: 50^M
The temperature in Celsius is: 10.00 ^M
^[]0;ryan@ryan-ece373: ~/ECE373^G^[]01;32mryan@ryan-ece373^[]00m:^[]01;34m~/ECE
373^[]00m$ exit^M

Script done on 2021-04-14 08:33:45-07:00 [COMMAND_EXIT_CODE="0"]
```

ECE 373 Assignment #1
Spring 2021

Source file:

```
#include <stdio.h>

int main() {
    float temp;

    printf("Enter temperature in Fahrenheit: ");
    scanf("%f", &temp);

    temp = (temp - 32) / 1.80;

    printf("The temperature in Celsius is: %.2f \n", temp);

    return 0;
}
```

D. Hello, Kernel:

Module code:

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

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_AUTHOR("Ryan Nand");
MODULE_VERSION("1.1");
MODULE_LICENSE("GPL");
module_init(hello_init);
module_exit(hello_exit);
```

```
[Wed Apr 14 08:25:52 2021] Hello Kernel
[Wed Apr 14 08:28:12 2021] Goodbye Kernel
ryan@ryan-ece373:~/ECE373$
```