ECE 373 – Assignment #3
Character Driver with ioctl

Check all of these materials into your Github Classroom repo before **11pm, Pacifc time, Wednesday, 8-May-2024**

Objectives:

This is a simple assignment to extend your char device driver from Assignment #2 to include *ioctl* functionality.

1. Setting up the char device (60 pts):

The first part of this assignment deals with introducing the *ioctl* functionality in the driver. In this part you will add an *ioctl* handler routine to the character driver. Assign a "*Magic Number*" to your driver and define two different operations *types*: Read and Write. Use the macros provided in <*linux/ioctl.h*> for constructing the commands to read and write data from the char device. Your ioctl handler should implement Read and Write functionality on the device depending on the command from the user.

2. User test code (40 pts)

Modify your C program in userspace from Assignment #2 to take an input from the command line and send/write to the device driver using *ioctl*. It should then read the value back from the driver using a different *ioctl* command and print the value to the console. You can use the <*linux/ioctl.h*> provided macros to build the command in the similar fashion like the driver.

Make sure your module unloads successfully, and properly cleans up ALL resources you allocated in the driver (`chardev`, `dev_t`, etc.).

What to turn in:

1. Source code to your kernel module, plus your kernel module Makefile.
2. Your userspace program. This does not require a Makefile, just the source code.
3. A typescript of loading the driver and running your userspace program.
4. A typescript of `/proc/devices` showing your module loaded.