Report for Programming Assignment 1:

The goal of programming assignment was to write a device driver for PCI device (software emulated device). Device driver is implemented as a LKM.

Design: Device driver is implemented as Loadable Kernel Module(LKM). Like other modules, our module for device driver also has an initialization and exit method. Initialization method invokes PCI driver registration function that registers our device driver for specified device(s).
All those initialization and registration are done by probe function like enable PCI device, allocation and mapping of device memory, registering irq etc. We are creating a character device named cryptocard that will show up as : */dev/cryptocard* . This is device file for our PCI device. This is the software interface through which we will interact with device. For this device file, various functions are defined like open, release, read, write, ioctl and mmap. When user program calls files related APIs from user space, then these functions are ultimately called and perform the requested operations as per user request.

Implementaton Details:

My PCI driver structure is cryptocard and I am globally maintaining some device and driver related informaton in cryptocard_mm struct like bar memory pointers etc.
I am maintaining FD specific configuraton setting in file->private_data.
I am using ioctl function to implement set_key, set_config , encrypt and decrypt functions. Through ioctll only i calling different functions for different configuration setting like encryption_dma_poll, decryption_mmio_interrupt, mmio_mmap_polling functions etc
I am controlling the more than 1 MB Size from crypter.c only.
I have slightly modifed crypter.h for ioctl and please use my crypter.h file only for evaluation.

To implement multiprocessing functions, i am taking spin_lock cc_spin_lock at time of encryption and decryption so that only one operation is allowed at a time.

My driver is passing all test cases except generic one .

For testing at development time, i created multiple user program to test the functionality.