Assignment 4
Documentation

Program Logic / How the Program Works:

1. In this problem, we were supposed to create a kernel driven producer and consumer.
2. So firstly, I wrote the two syscalls, writer and reader, in the /kernel/sys.c file of the Linux kernel.
3. I also had to make a work queue so that the producer can write to it and the consumer can read from it.
4. Then I made semaphores namely empty, full and a mutex to avoid racing. I also initialised two file descriptors write and read.
5. Now for the consumer to delete/read from the queue I made a delete function that took care of specifically the dequeue operation. And similarly, I made an add function for the producer to add/write data to the queue which basically performed the enqueue operation.
6. The reader and writer syscalls took use of the add and delete functions to read/write the work queue.
7. After the work on sys.c was done, I added the newly made syscall in the /arch/x86/entry/syscalls/syscall_64.tbl.
8. I now compiled the kernel to integrate the newly added syscalls and then made two c files namely, producer.c and consumer.c to test the syscalls.
9. The producer reads random 8-bytes of the device /dev/urandom and passes them to the kernel using the writer syscall. The consumer uses the reader syscall to read the random number written by the producer.

How to compile the program:

1. First, apply the patch file provided to the stock kernel.
2. Now run the make command on the terminal to produce the consumer and producer executable.
3. Open two terminals and run ./producer in one and ./consumer in the other.