Question 3: Message Queues (6 pts)

Review the programs (spock.c and kirk.c).

Answer (or discuss) questions listed below:

**a) Discuss and evaluate what happens when you're running both in separate**

**windows and you kill one or the other.**

If you kill Kirk first via “kill -9 <pid>”, then nothing happens. Spock continues to run on normally, although typing anything in the Kirk terminal will no longer be sent to and printed out on Spock. If you quit using the ctrl + D command in Kirk, Spock in the other terminal will also be terminated.

If you kill Spock first and then continue to type in send messages from Kirk, nothing will happen. That is until you run another instance of Spock, in which all of the messages you typed in Kirk while Spock was not running will be printed out immediately.

This is most likely because Kirk bundles every message together into a queue and then Spock reads that queue. When A new Spock starts up, it reads in all of those queued messages at once.

**b) Discuss what happens (and why) when you run two copies of kirk.**

When two copies of Kirk run, you can type in both terminals and ‘send’ messages, but since there is no Spock to receive and print messages. Spock is the only that has a statement for printing the buf.mtext, so of course if two Kirks run, messages will be sent but not printed/received. If you open a Spock later on, it prints out all queued messages from both Kirks in the order they were sent.

**c) Discuss what happens (and why) when your run two copies of spock.**

When two copies of Spock are ran, nothing is able to be done. Both are waiting for messages, but neither Spock can send any. They never quit either. The ability to send messages is tied behind the Kirk program**.** I assume this is because the Kirk program is the only one that uses IPC\_CREATE as a flag, meaning Spock doesn’t have the ability to create message queues.