Project 2
Application for Multithreaded Synchronization

Dustin Nguyen
016873907
Katherine Seng
016590715

For the project, we wrote a C program using multithreaded synchronization to implement an office hours scenario between a professor and a select number of students in a room with limited capacity. The number of students and capacity of the professor's office are designated by integer values given in the command line of the terminal. These values are validated within our code and will print out a message if it is determined that the user did not enter non-decimal numbers in the command line.

Our code contains multiple functions that were designated in the project instructions, mostly using mutex locks to synchronize the threads properly. Creating the code, most of our time was spent resolving errors. This is because we chose mutex locks instead of semaphores for our project. We struggled specifically with a situation in our code where the professor doesn't answer a student's question in the office, but this was resolved after correcting our code to use certain locks for certain events (i.e. *q_mutex* for questions and *a_mutex* for answers).

```
glassheart@KatherineSeng-VirtualBox:~/Desktop/code$ ./run 2 3
Student 1 enters the office.
Student 1 asks a question.
Student 0 enters the office.
Professor starts to answer question for Student 1.
Professor is done with answer for Student 1.
Student 1 is satisfied.
Student 1 asks a question.
Professor starts to answer question for Student 1.
Professor is done with answer for Student 1.
Student 1 is satisfied.
Student 1 leaves the office.
Student 0 asks a question.
Professor starts to answer question for Student 0.
Professor is done with answer for Student 0.
Student 0 is satisfied.
Student 0 leaves the office.
```

Figure 1: Output of the program assuming two students and a room capacity of three

We also created a Makefile and a respective README.txt for this project.

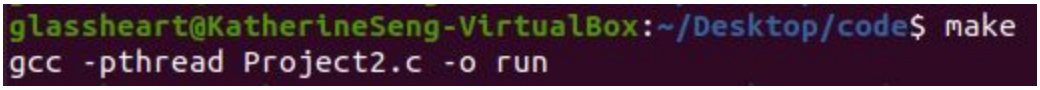

Figure 2: Makefile set-up as indicated in the README.txt

For this project, members Dustin and Katherine collaborated in code-level discussion with each other. As with the previous project, we tried to designate a leadership role for different aspects of the project and therefore had our own primary responsibilities. Dustin wrote the majority of the C program. Katherine assisted with coding, as well as working on all of the other deliverables needed, including the project report.