**Design of Sleeping Teaching Assistant**

Members: Emmanuel Khlot, Marco Botello, Sahil Sheth, Cesar Martinez Melgoza

1. **Problem Statement**

We were tasked with using our knowledge of pthread and

process/thread synchronization in order to create a program that emulates a Teaching Assistant helping several students. This included use of semaphores and mutex locks.

1. **How to use your program**

To use this program, you must first locate the folder titled Sleeping-TA-master. Within this folder you will find the TA.cpp file which would then need to be compiled using the “g++ TA.cpp -o main” command in the terminal. This creates a main executable which you can run using the command “./main” in the terminal.

1. **Design of your program**

We used several semaphores to represent students and one to simulate the TA. The TA receives a signal from a student in the first seat and is woken up to help the student. If another student tries to signal the TA while it is helping another student it will ignore it and the student will be forced to wait. Once the TA is done helping, the student leaves and the next student in line enters the TA’s office for help. If there are no students who need help, the TA will continue to sleep.

1. **Any Limitations**

There were no limitations set in the application.

1. **Any shortcomings**

One shortcoming we were unable to find the cause of was the chairCount. It seems the way we chose to calculate the chairs left when a student takes a seat is not always correct. At times the program will even state that there are -1 chairs left.

