Dianne Lopez
Kailie Chang
Heriberto Juache
CPSC 351-01
11/12/19

Design of Sleeping Teaching Assistant

Modularization

Two Modules:
- Teaching Assistant (TA): waits for the first student waiting to wake him/her up, helps the student, releases the student when he/she is done helping them, and signals the next student waiting to come in.
- Students: signals the TA to wake up, occupies the chairs when there are other students waiting, waits for their turn to get help, leaves his/her chair when they are called in, and leaves if the TA is done helping them or if the available chairs are filled.
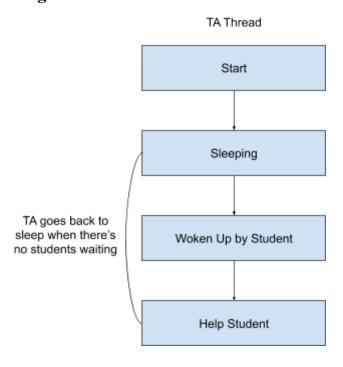
Concurrency

In this application, the student and the TA are the modules that are executing in parallel. The TA is executed first and begins asleep since there are no students in the chairs that need help. The student then wakes up the TA to receive help during office hours. Doing this, concurrency of the modules allows both TA and Student to be executed parallel to each other to ensure that the information Students are carefully being helped one (thread) at a time.

Coupling and Cohesion

The cohesion between the TA and Student is very high since both threads need to work together for it to work properly. The student threads have to be in sync with the TA thread at all times, since both rely on each other for it to actually run. Since there is only one TA thread running and three student threads, the student threads will signal the TA thread on what to do. For instance, if a student needs help and the TA is asleep, the student will signal the TA to wake up and help the student. If there are no students that need help (e.g. all chairs are available), the TA will go to sleep.

The coupling between the TA and Student modules are very high. This means that they have a strong relationship with one another and dependent on each other. Since the Student module decides whether the TA is asleep, this is considered Control Coupling.

Design Verification

TA Thread

Student Thread