Jay Gallardo
Leonel Noriega-Rojas
Noemi Pantoja-Morales
Rith Sreng
CPSC 351: Operating Systems
11/17/2024

Design of Sleeping Teaching Assistant

Problem

A university computer science department has a teaching assistant (TA) who helps undergraduate students with their programming assignments during regular office hours. The TA'S office is small and has room for only one desk with a chair and computer. There are three chairs in the hallway outside the office where students can sit and wait if the TA is currently helping another student. When there are no students who need help during office hours, the TA sits at the desk and takes a nap.

If a student arrives during office hours and finds the TA sleeping, the student must awaken the TA to ask for help. If a student arrives and finds the TA currently helping another student, the student sits on one of the chairs in the hallway and waits. If no chairs are available, the student will come back at a later time.

Definitions

Constants:
MAXCHAIRS: the maximum number of chairs available for students waiting for help (3)
pthread_t *Students: create multiple threads for students
pthread_t TA: create single thread for TA
ChairsCount: track the number of occupied chairs
CurrentIndex: current chair index
studentsLeft: total number of students left needing help

Semaphores & Mutexes:
TA_sleep: used to wake up the TA when a student arrives
Student_sem: used to indicate that a student has received help and will be leaving soon
chairs_sem[MAXCHAIRS]: an array of semaphores, one for each chair, that is used to coordinate each student waiting in line
mutex: a mutex lock to prevent race conditions when students access and modify ChairsCount

Functions

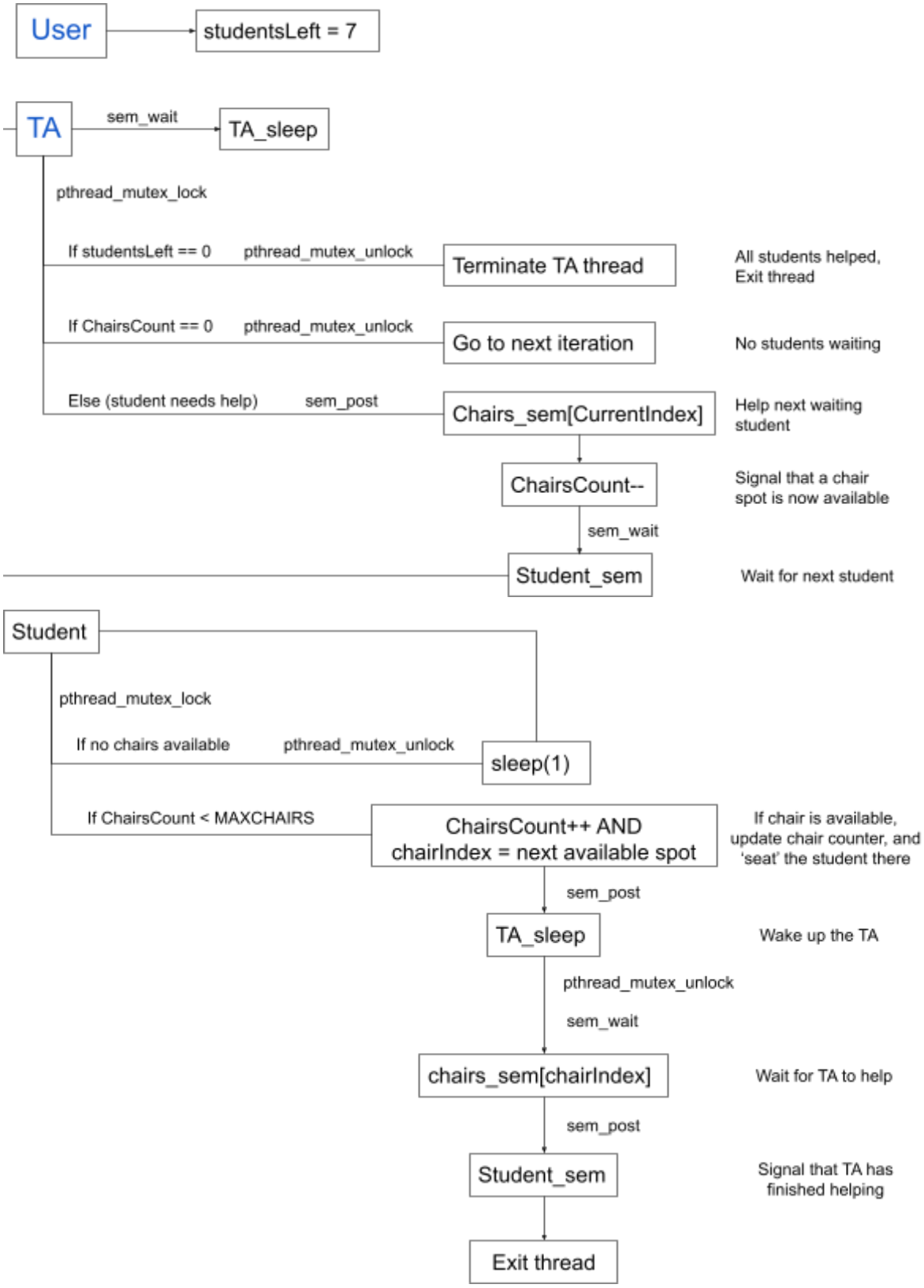
Main: The main function takes the input provided by the user to set the variable number_of_students. If no input is provided, the default value of 5 is used. The mutex and

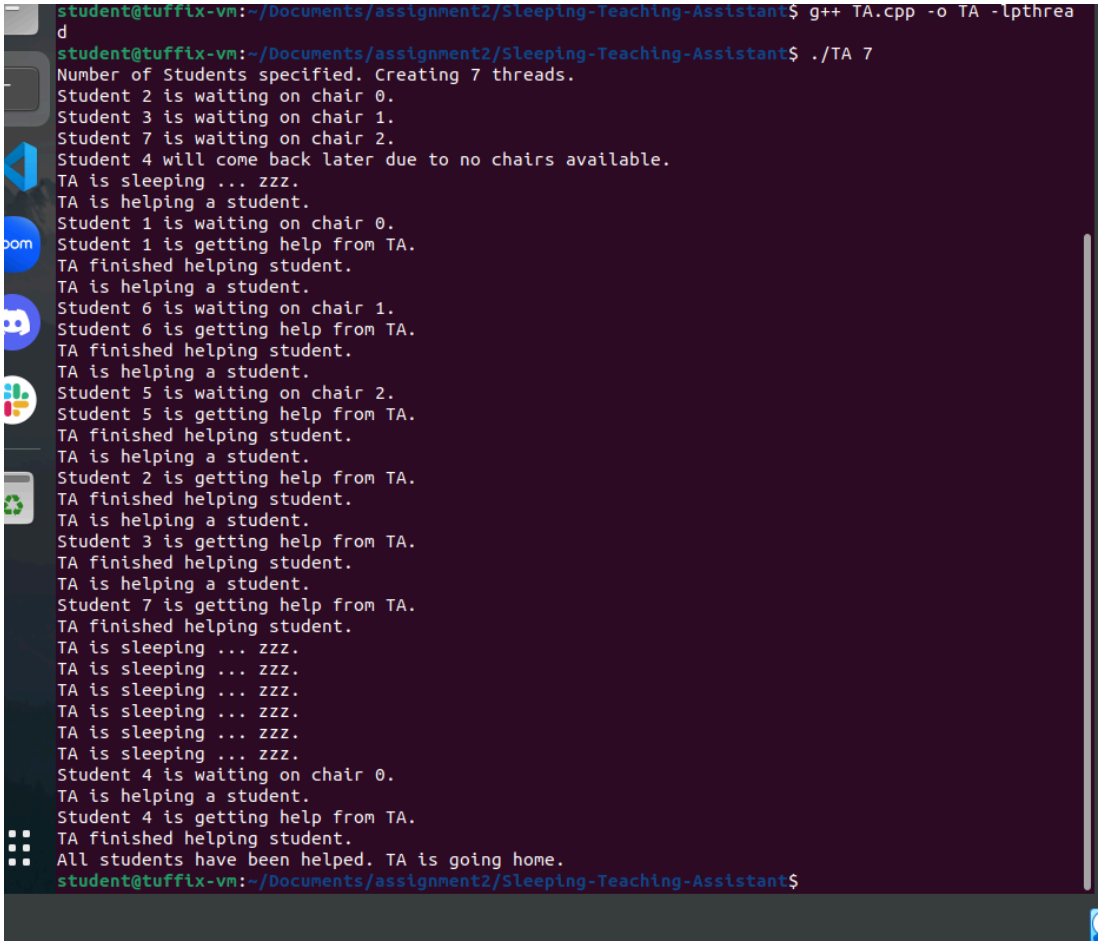
semaphores are initialized. Then, the function allocates enough memory for the Students thread and creates a thread for each student and one for the TA. pthread_join is used to wait for all threads to complete. Once the threads are completed, the function deallocates memory from the Students thread and destroys all semaphores and mutex. The program terminates.

TA_Activity: This function puts the TA to sleep until a student arrives. When the TA is woken up by a student, the TA checks if any students are still waiting. The TA helps the student on the next available chair, cycling through the chairs as a bounded buffer. After helping a student, the TA signals that the student can leave and decreases the studentsLeft count. If the TA has helped all students, the TA is done for the day and the function will exit the TA thread. If there are empty chairs, the function breaks the loop and the TA takes a nap.

Student_Activity: Each student thread attempts to get help from the TA. If a chair is available, the student sits on a chair, increments ChairsCount and wakes the TA if sleeping. If no chair is available, the student will come back later and try again. Once the student receives help from the TA, the function completes, ending the student thread.

Flowchart/Visualization:




student@tuffix-vm:~/Documents/assignment2/Sleeping-Teaching-Assistant$ g++ TA.cpp -o TA -lpthrea
d
student@tuffix-vm:~/Documents/assignment2/Sleeping-Teaching-Assistant$ ./TA 7
Number of Students specified. Creating 7 threads.
Student 2 is waiting on chair 0.
Student 3 is waiting on chair 1.
Student 7 is waiting on chair 2.
Student 4 will come back later due to no chairs available.
TA is sleeping ... zzz.
TA is helping a student.
Student 1 is waiting on chair 0.
Student 1 is getting help from TA.
TA finished helping student.
TA is helping a student.
Student 6 is waiting on chair 1.
Student 6 is getting help from TA.
TA finished helping student.
TA is helping a student.
Student 5 is waiting on chair 2.
Student 5 is getting help from TA.
TA finished helping student.
TA is helping a student.
Student 2 is getting help from TA.
TA finished helping student.
TA is helping a student.
Student 3 is getting help from TA.
TA finished helping student.
TA is helping a student.
Student 7 is getting help from TA.
TA finished helping student.
TA is sleeping ... zzz.
TA is sleeping ... zzz.
TA is sleeping ... zzz.
TA is sleeping ... zzz.
TA is sleeping ... zzz.
TA is sleeping ... zzz.
Student 4 is waiting on chair 0.
TA is helping a student.
Student 4 is getting help from TA.
TA finished helping student.
All students have been helped. TA is going home.
student@tuffix-vm:~/Documents/assignment2/Sleeping-Teaching-Assistant$