Alexandria University
Faculty of Engineering
Comp. & Comm. Engineering
CC373: Operating Systems
Fall 2018

جامعة الاسكندرية
كلية الهندسة
برنامج هندسة الحاسب والاتصالات
مادة نظم التشغيل
خريف ٢٠١٨

Lab3
Mutual Exclusion and Synchronization
Train Automation

Objectives:

1. To understand mutual exclusion and synchronization techniques
2. To work with Mutex Variables in Pthreads.

Introduction:

We have decided to improve the train efficiency by automating not just the trains but also the passengers. From now on, passengers will be robots. Each robot and each train is controlled by a thread. You have been hired to write synchronization functions that will guarantee orderly loading of trains.

Requirements

- You must define a structure `struct station`, plus several functions described below.
- When a train arrives in the station and has opened its doors, it invokes the function `station_load_train(struct station *station, int count)`, where count indicates how many seats are available on the train. The function must not return until the train is satisfactorily loaded (all passengers are in their seats, and either the train is full or all waiting passengers have boarded).
- When a passenger robot arrives in a station, it first invokes the function `station_wait_for_train(struct station *station)`. This function must not return until a train is in the station (i.e., a call to station_load_train is in progress) and there are enough free seats on the train for this passenger to sit down. Once this function returns, the passenger robot will move the passenger on board the train and into a seat (you do not need to worry about how this mechanism works).
- Once the passenger is seated, it will call the function `station_on_board(struct station *station)` to let the train know that it's on board.

Create a file train.c that contains a declaration for struct station and defines the three functions above, plus the function `station_init`, which will be invoked to initialize the station object when the system boots.

Notes:

- You must write your solution in C using Pthreads and its Mutex Variables.
- You may not use more than a single lock in each struct station.
- You may assume that there is never more than one train in the station at once, and that all trains (and all passengers) are going to the same destination (i.e. any passenger can board any train).

- Your code must allow multiple passengers to board simultaneously.
- Your code must not result in busywaiting.

Deliverables:

- Complete source code, commented thoroughly and clearly.
- Object Code
- A report that includes:
  - A description of the overall organization of your code and the major functions
  - Sample runs
- You should work individually.
