

Assignment 4: Street Simulation

CS-232 Operating Systems
Spring 2022

Due: 11:59pm Monday, 28th November 2022

1 Overview

In this assignment, you will synchronize a multi-threaded C application that simulates traffic on a street. The primary goal of this assignment is to apply pthread synchronization primitives such as mutexes and conditional variables to solve the synchronization problem. Please use discussion thread on canvas for any concern related to the assignment.

2 Instructions

Read the accompanied `traffic.c`. It takes an input file as argument and creates multiple threads, one for each car plus one more for the street itself. The input file can contain any number of lines, each denoting an entry for a car. Each entry contains the time of arrival since the previous car, the duration of travel and the direction which the car is going into.

Our current `traffic.c` allows no communication so the threads are working independently. Your task is to achieve communication between the threads via synchronization variables such that:

1. No more than 3 cars should be present on the street at any instance; let us say it is illegal. Subsequent cars can only wait until any one of the 3 cars leaves.
2. We assume the street is too narrow for cars to be incoming and outgoing simultaneously. So there will be either only incoming cars or only outgoing cars travelling at a time. However, the streaming should keep running without deadlock in either direction irrespective of how cars arrive.
3. After every 7th car leaves, the street becomes unusable and has to be repaired. Cars do not enter the street unless it is ready to use. Only the street thread is allowed to repair the street.

You can only add your code where indicated. We do not want the trivial solution (0 marks for this one) which eliminates concurrency entirely and sends only one car at a time. So, your solution must maximize concurrency. Note that you do not have to respect the ordering of every event and there can be multiple valid simulations for the same input as long as the above rules are satisfied.

3 Submission

Your submission will comprise of the modified `traffic.c` and a document (no more than 2 pages) explaining why your logic fulfills the given criteria and ensures maximum concurrency. **Do not change any print statements in your final submission.**

4 Rubric

1. No more than 3 cars travel [20 marks]
2. Travelling cars at any time have the same direction [20 marks]
3. The street repairs after every 7th car [20 marks]
4. Maximum concurrency allowed (given that rules are satisfied) [20 marks]
5. Document is to-the-point and justifies all requirements [20 marks]