**CSC 360 p2a**

**Naum Hoffman**

**V00927502**

1. How many threads are you going to use? Specify the work that you intend each thread to perform.

I plan to use one thread for every train specified in the stdin. There will also be an additional thread, which will be used to dispatch trains, and a controller thread in the main thread (main function)

1. Do the threads work independently? Or, is there an overall “controller” thread?

In the main thread, there is an overall controller, so no, they don’t work independently

1. How many mutexes are you going to use? Specify the operation that each mutex will guard.

There will be 3 mutex, for each station and the dispatcher thread.

1. Will the main thread be idle? If not, what will it be doing?

My main thread acts like a controller thread, so it will not be idle

1. How are you going to represent stations (which are collections of loaded trains ready to depart)? That is what type of data structure will you use?

I will use an array, which will be dynamically allocated, to store the loaded trains in order of low to high priority.

1. How are you going to ensure that data structures in your program will not be modified concurrently?

I will guard the data structures in my program with mutexes, making sure that the trains don’t “leave” simultaneously.

1. How many convars are you going to use?
   1. Describe the condition that the convar will represent

For every train, there will be two convars, one for loading time and one for crossing time.

* 1. Which mutex is associated with the convar? Why?

The station mutexes will be associated with the loading time convar, ensuring that they can’t all “load” at the same time.

* 1. pthread\_cond\_signal(convar)

1. In 15 lines or less, briefly sketch the overall algorithm you will use. You may use sentences such as:

If the train is loaded, get station mutex, put into queue, release station mutex

In load function:

Initialize station mutex

Load train

If train is loaded, get station mutex, put into array, sort array, release station mutex

Print train loaded

In Dispatch function:

Initialize dispatch mutex

Using additional array, sort east station and west station trains

Get dispatch mutex, dispatch train according to east or west station in array, pop train from array when used.

Release dispatch matrix.

In main:

Parse stdin file and initialize station arrays

Load()

Dispatch()