

CS 4348 Project 2 Summary

The purpose of this project was to simulate a theater with semaphores. I decided to implement the project in Java. I separated each of the different types of threads into their own Runnable classes. The main method then appropriately instantiates and runs them all. Communication between the customers and any of the employees follows the same structure: First the employee accepts the first customer to acquire its semaphore. Then the customer stores information it wishes to pass to the employee, usually an ID or index of some sort, in a global signals the employee that it has made its request and waits for it to finish its job. The employee then begins doing the job it is supposed to do using the ID in the global variable, and finally signals the customer to continue to the step. The employees repeat this process for all customers.

It was fairly difficult in the beginning during the planning stage. I could not come up with a satisfactory method of transferring data between threads. The method of storing data in globals and coordinating access with semaphores I use works but feels very messy. Also, I'm not entirely sure that my program has any mutual exclusion; I thought that would have been necessary, but it seems that it isn't. Ignoring that, once I figured out the structure for thread interactions I explained above, it was fairly easy to design the program.

After I wrote down the pseudocode of the design of the program, writing the program was fairly straight forward. In fact, I did not run it until I had written the entire program, and when I was done it worked perfectly the first time I ran it, at least from what I can tell. It worked fine for a simulation of three customers. I ran it with a simulation of 300 customers, and at a

glance, the simulation output seemed to be correct, but it is difficult to completely verify that it is exactly correct.