The simulation creates 2 box offices to handle the 50 customers. The 50 customers pick a random movie, if the movie has still available, they buy a ticket, if not they leave the theater and the thread ends. After buying the ticket, the customer has a 50% chance of going to the concession stand to buy either popcorn, soda, or both. After going to the concession, they go to the ticket taker where the ticket taker lets them in the theater and the thread ends.

The main difficulty I encountered was transferring the code to Unix. I used drjava to run and test my code because it was much faster to compile and run compared to doing it in Unix. Once it worked in drjava, I moved it to Unix which kept on getting Null Pointer Exception error. After waiting for a day, the program just seemed to start working on Unix with little to no problems which still confuses me.

The other problem I encountered is that when the customer leaves the theatre because movie seats where not available, the t\_Buy semaphore would not increment because only the box office has the power to increment t\_Buy. I solved this issue by incrementing the t\_Buy when movie seats are not available or when the box office gives the movie ticket the customer wants.

The last problem I had is that the “Theatre is open” is printed before the concession stand worker and the ticket taker is open. I tried to create a semaphore to print it after the everything else is created but it would just create a NullPointerException error. After trying a few other ideas, it still wouldn’t work. So I kept it as it is.

The results of the program are almost the same as the sample output. 90 percent of the time, the code works without any problems. But the other 10%, the code will have a NullPointerException on the first or second customer thread which breaks the program. I have not been able to solve this problem. Another thing different is that the “Theatre is open” is sometimes printed before concession stand worker, ticket taker, and box office is open.