Rochester Institute of Technology

Real Time and Embedded Systems

Project 4 – Bank Simulator Design Document

March 26, 2017

Zachary Weeden

To simulate the queue of the customers arriving, a queue/array-like strucutre could be put into place in it’s own thread with elements appended/dequeued. These elements in this array of the thread are representative of customers and each element can simply be the numeric representation of the duration of the customer’s transaction to be taken place. This number is to be randomly generated as specified in the documentation.

Each of the three bank tellers are to have their own threads where they can individually handle and process their customer – pulling from the customer queue//thread. On accepting of a customer, that particular thread should “lock” and thusly not accept any more customer’s until that transaction has completed. A non-blocking wait for ‘x’ amount of time will be implemented in the corresponding thread for a varied time depending on the randomly generated time for the customer. Research on mutex’s and semaphores is needed and believed to be implemetned here.

After the non-blocking timer has completed and the top value is met/exceeded (the customer’s randomly generated set time), the customer is then considered to be taken care of and the thread will then “unlock”(this is to say the teller is now available) and accept the next customer in the customer queue/thread for their respective set time, resetting the non-blocking wait and adjusting the top for that customer appropriately.

An overarching timer will be needed to simulate/handle the bank’s hours of operation and further research on how to manipulate thread’s interaction and behavior with eachother is needed

Isolation of the number crunching (statistics, results and overall reporting) will be isolated in its own file outside of main.