Erik Poole

1) Refactoring -

There are numerous sections in my simulator code that could benefit from refactoring, but my main is definitely the biggest offender. Currently the code is almost completely done in duplicate for testing the market and bank queues. The first step will be to rework that into a function that can be reused for each case. I'm also currently generating my priority queue of arrival events twice in main, and it would be worthwhile to rework things so that I only generate things once and then pass the queue by value to be used independently by both the bank and market. My code for printing the results are already partially broken out into a function ("printPercentiles") but there is still a lot of reused code there that could be cleaned up as well. The printPercentiles function is currently part of my event.cpp file, which is also probably not a great place for it.

I think I'm overall happy with my event class and its subclasses "newPersonEvent" and "TellerFreedEvent", but I would like to reorder and perhaps rename the arguments passed to their "handleEvents" method. I'd also like to think of a better way to hand line switching than to have every handleEvent method pull in the entire vector of lines to do math with, maybe by using method overloading for different event types.

Right now my line class is functional, but I'm exposing an uneven interface. When I add a customer I have a method for it, but when I remove a customer I am directly accessing the queue from outside the class. I should add a "remove customer" method to hand that more consistently.

For all of my classes all member variables are currently publically available and most don't need to be, or at the very least I should think more carefully about whether they need to or not. Adding appropriate accessor functions for private variables I do need to access somewhere else is also probably a good thing to work on.

Lastly, I have a lot commented out lines that were used to print information for debugging that is no longer particularly useful. I should either remove it or set it up to only be utilized when a certain compiler flag is used.

2) Surface Plots -

Both of my surface plots appear to portray the same dataset because my model output identical metrics for both the Bank and Market simulations. The area depicted by the graph is where the time taken dramatically increases as the lines become oversaturated.