Jesse Narkmanee

805190133

CS 31 Project 5 Report

**Encountered Problems**

While testing my program the only, the first problem that I encountered was my Auction bid function not working since for some reason it was not updating the member variable mCurrentBid. However, I was quick to find the fix in that I assigned the amount parameter to mCurrentBid instead of the other way around, a careless mistake. A second problem included the last assert case given in the spec where unsuccessful auctions should not raise the earned award in addAuction. I found my problem to be that the function winingBid() returns -1 if the auction is unsuccessful so needed to write a case in addAuction() to make sure that we do not add on to EBayBucks earnedAward. I went ahead with a simple if else statement and the program now works successfully since it now includes cases when auctions are both successful and not successful in applying EbayBucks earnedAward.

Lastly, while not a huge problem, in the beginning it was hard to visualize the big picture in what the spec was asking me to perform for this project. What I did to remedy this was first to write all the function prototypes in the header file then respective .cpp files and try to make sure to link them properly. Then I pasted each line of spec describing each action to its respective function that I thought it applied to. Then having these comments separated and above their respective function prototypes gave me a good idea of how to implement the project as a whole since I would follow each sentence or two at a time enabling me to engage in solving the sub problems eventually solving the entire program.

**Test Cases**

Test Cases I ran included things the cases found on the spec as assert statements and test that ran on Codeboard. All these tests would be successful. After that, I ran my own tests that included when the bid is still open then there should not be a winning bid and winingBid() should return -1. Also, I tested negative number to throw false bids and if I passed in larger priced bids that this would not break my code as my program would still correctly determine number of bids and which bids are considered to be the highest bids. This would also be successful. Lastly, when I close the bids, I want to make sure that no more bids would go through and that the number of bids would not change as well.

My winning bid would turn out to be 4310.00 and now I wanted to see if earnings at 1% would be calculated correctly after casting it to a string type. This would indeed be successful again with both earnings and the certificate showing the same correct amount in addition to the certificate resetting the eBucks earnings after it is called.