Jesse Narkmanee

805190133

CS 31 Project 6 Report

**Encountered Problems**

I encountered a couple of problems that did take a pretty long time to solve on Codeboard. First, my FIVETIME multiplier test kept on failing and I did not really know why since I could not see what the test case was and what it was outputting on Codeboard. However, I eventually inserted a cout right before the return statement and it outputted on Codeboard that it was returning 10 (first case) instead of 5, which lead me to the problem right away in that my if statement was not having all three different types of wheel by instead having the third wheel being listed as the first still.

My second encountered problem that did take a long time again to figure out was why I seemed to be failing a test case in it creating a wager win that was a little lower that what it should have been. After eyeballing my code for awhile thinking it was in my cpp implementation, I realized that the problem was in the multiplier header file in that the enum data types were not mapping to the correct number for TENTIME. Initially, I thought that maybe I would have to use some type of switch or if-else but realized that I could just auto-initialize it in the header file which worked perfectly.

**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 pass. After that, I ran user input test cases using the default main supplied and inputted some numbers that I believe would be encountered since the specs mentioned them. First, I made sure to test whether wager limits would not exceed what was in the bank and that bank balances would update accordingly as I deposited more money. All these test cases would pass.

My next test on the user i/o would be to see whether cashing out different amounts would reset my bank balance which it successfully did. Next, I wrote assert test cases for every single multiplier to see if the cheat play function would work if the wheels hit an expected multiplier. All these test cases would pass as well. After that, I tested to see whether some of my individual functions would work together correctly with the cheat play function such as playing with different wheel combinations and seeing if the cash out functions would work and me adding some other functions after such as the win and lose function would update my balance accordingly. All these test cases would work successfully.