Ann Kostecki
CS 362-400
7/22/18
Assignment 3

Bugs

For each of the card unit tests, I checked various items within the Game state to see if they changed as expected. When testing Smithy, that had known bugs, I found that many values differed, including the number of cards in the hand and in the deck. Because I knew what bugs I introduced, these results were expected and matched what I believed would happen. However, for someone who would not know about the bugs, the mismatch in counts and expectations would provoke further research.
Similar to the Smithy card, the Adventurer and Council Room cards also presented discrepancies between the values that were expected and what was actual. Both of these cards also had bugs introduced in their functions that caused the functions to turn out different than expected.
The Village card did not have any bugs introduced in the code and the results for the counts matched what was expected. In testing the compare function and updateCoins function, all tests passed and no bugs were found.

Unit Testing

For each of the card tests and unit test 2, I had 100% coverage. For unit test 1 (compare), I found that the lines that did not run were print statements if the testing failed. In considering this, I would modify my testing to intentionally include tests that would fail.
Coverage for dominion was at about 50%. This is somewhat to be expected, since I did not test all lines of code within the file. In reviewing what I did test, I found that almost all lines were covered. Though initializegame() was not tested, I saw that not all lines were covered. This is because when I used this function, I used the same cards to keep the rest of testing consistent. If I were to test different cards or test with more variety, I would initialize with different game cards in order to produce 100% coverage. The functions that were used for the card testing showed 100% coverage.

Unit Testing Efforts

When unit testing, I carefully check through which variables are included and expected to change. I make note of these variables so that I am sure to cover their various states in testing. While card testing, I tested all the same variables throughout. I feel that this is important so that I am not only testing variables that I know are going to change but also variables that shouldn't change. If a variable changes that shouldn't, this is also represents a bug, just as if there was a variable that didn't change that should (or changed in a way that wasn't expected).
After I identified the variables to test, I considered in what ways they could change. For example, if a variable was an integer, the ways it could change would be to increase, decrease or stay the same. As well, with integers, I wanted to consider if the value could be negative. With one of the tests, I purposely made the result a negative integer, when in reality, the game should not allow the value to be negative.