Benjamin Tate
10/8/2017
CS 362 -- Section 400
Assignment 2

Refactor:

I chose to refactor the same 5 cards that I used for assignment 1: smithy, adventurer, baron, feast, and minion. For each of the 5 cards, I found the corresponding switch case in the cardEffect() function and manually scanned the case for variables that would need to be passed to the function or declared within the function. I used this information to create a function prototype called *cardName*Case() and add the necessary variable declarations to the top of the function. Finally, I cut all of the lines from the switch case up to “return 0;” pasted them into the new function, and replaced them with a call to that function within the switch case.

After completing each function, I tested to make sure I didn’t miss any variable declarations or passes by running “make dominion.o” and making sure it was successful. Once I had refactored all 5 of the functions, I commented out unused variables in cardEffect() to get rid of compiler warnings and cleaned up the indentations.

Bugs:

I introduced bugs in the smithy, adventurer, baron, and feast case functions. In each case, I added a comment to the end of the buggy line that says “//BUG (*bug specifics*).” The following bugs were introduced:

1. Smithy: for loop changed from i=0-3 to i=0-2, so that only 2 cards will be drawn instead of 3.
2. Adventurer: first while loop changed from drawntreasure<2 to drawntreasure<3, so now 3 treasure will be drawn instead of 2.
3. Baron: card_not_discarded is now initialized as 0 instead of 1, so the while loop within the first if statement will be skipped.
4. Feast: in the last else statement, the line assigning 0 to x was commented out, which will create an infinite loop.