Megan Weller
ESOF 322 – A3
Executive Summary

1. Testing Approach

My testing approach was compare whether the method performs what I am expecting it to. For example, if the user is entering a room, I am expecting the user to be in Room 2 after the method is completed. I then test the returned value with my expected value. For all tests, I either used assertEquals() or assertTrue(). For methods with if-statements, I created a specific amount of tests to cover all the branches in the if-statement.

Tools:

- ECLEmma
- JUnit4

2. Number of tests performed in each test case:

Door enter() – 4
Player go() – 1
Player pickup() – 2
Player drop() – 4
Room addItem() – 1
Room removeItem() –1
Room enter() – 1
Room exit() – 1

3. Test Coverage

- Door enter() – 100 % instruction coverage – 83% branch coverage

  I am missing 1 of 2 branches on the line:

  else if (p.getLoc() == inSite){ ….. }

  This is when the player is in Room 2 and is using the key to get into Room 1. I tested it when the Player had the key and entered the door and when the Player did not have the key and tried to enter the door. I do not know of another way to test this line since the room can either be assigned outsite or insite and the only way to get to the else if() line is for p.getLoc() to be insite. There is not a third case that I can see that would not execute the else if() statement.

- Player go() – 100% instruction coverage – n/a branch coverage
- Player pickup() - 100% instruction coverage – 100 % branch coverage
- Player drop() - 100% instruction coverage – 89% branch coverage

There is a switch statement that is missing 1 of 3 branches. It does cover both case1 and case 2, but I am assuming the third branch is a missing default case to catch a fall through. I am not able to test a default case therefore I will mark this as an error.

- Room addItem() - 100% instruction coverage – n/a branch coverage
- Room removeItem() - 100% instruction coverage – n/a branch coverage
- Room enter() - 100% instruction coverage – n/a branch coverage
- Room exit() - 100% instruction coverage – n/a branch coverage

4. There were 0 failures within the methods that were assigned to test.