EE 422C March 6, 2016
Brandon Arrindell, Cooper Raterink

Testing

Word Ladder Testing	
fiveLettersTest()	validateSimpleLadderTest()
notAWordTest()	validateStartEndTest()
nullLadderTest()	validateInvalidWordsTest()
noSuchLadderTest()	validateChangesTest()
easyLadderTest()	validateDuplicateWordsTest()
hardLadderTest()	
trivialLadderTest()	
stressTest()	

Black Box Testing: The program takes in a dictionary file and an input file and outputs word ladders between inputs. Black Box testing assumes we have no knowledge of how the program operates, but do have knowledge of the inputs and expected outputs. To test the program as a "black box" we created a series of start words and end words, some with ladders and some without, and tested that our program operated as expected. This file is called *BasicTestCases.txt* in our project. These tests give us an idea of whether the program operates efficiently and correctly and that input error handling is working as expected (edge cases, bad inputs, etc.)

White Box Testing: White Box testing assumes we have full knowledge of how the program operates (its structure, what exceptions exists and when should they be thrown, etc). To test the full coverage of our program and all of it's parts, we created JUnit test cases to test some of the key functionality. In the table above you can see all of the test cases that exist. The table on the left tests the expected behavior of the computeLadder() and MakeLadder() methods. With the tests above, we are testing the main recursive algorithm for finding ladders, the HashMap creation of words and their candidates, and the validation method.

fiveLettersTest() - Tests that a NoSuchLadderExcpetion is thrown when input words are more or less than five letters.
notAWordTest() - Tests that a NoSuchLadderExcpetion is thrown when the one of the input word does not exist in the dictionary.
nullLadderTest() - Tests that NoSuchLadderExcpetion is thrown when the inputs are null.
easyLadderTest() - Tests that NoSuchLadderExcpetion is *not* thrown (AKA there is a ladder) between two similar words.
hardLadderTest() - Tests that NoSuchLadderExcpetion is *not* thrown (AKA there is a ladder) between two very different words.
trivialLadderTest() - Tests that NoSuchLadderExcpetion is *not* thrown when the start and end word are the same.
sterssTest - Ignoring NoSuchLadderExcpetions, this test creates ladders between the first 200 words in the dictionary, two at a time. Useful for testing reasonable time and ensuring that the program does not crash under heavy load.
validateSimpleLadderTest() - A correct ladder is inputted and we check that the validate method works.
validateStartEndTest() - Should not validate a ladder with incorrect start/end words.
validateInvalidWordsTest() - Should not validate a ladder with any invalid words.
validateChangesTest() - Should not validate a ladder with sequential words not differing by exactly one letter.
validateDuplicateWordsTest() - Should not validate a ladder with duplicate words.