Documentation:
FuzzyWuzzyTestDriver.py
Runs specified test cases from the testCases directory. Outputs relevent information to standard output. Can be run with, or without
arguments.
arguments: [nameOfTestFile] [test#]

function: argHandler(args)
Parses the arguments, gives one extra chance before stopping the program for bad input
input: <list>
output: <String> <int>

function: populateTestFiles(args)
Lists out all the files in the testCases directory.
output: <list>

function: parseTestFile(fileString)
Returns a dictionary containing the test cases for specified fileString
input: <string>
output: <dictionary={1: 'function' , 'inputValues', 'expectedOutput'}>

function: lineStripper(lineString)
Strips a line of tags, and outside whitespace.
input: <string>
output: <string>

function: runTests(test, testCase)
Runs test for given test case and test file. Outputs to standard out.
input: <string>, <string>

Test Case specification:
Test: <test identifier>
function: <function header>
input: <inputs, formatted as if they were being called by the function>
output: <expected output>

Test Cases:
Test:1
function: fuzz.partial_ratio
input: "","testing empty string"
output: 0

Test:1
function: fuzz.ratio
input: "","testing empty string"
output: 0

Test:1
function: fuzz.token_set_ratio
input: "","testing empty string"
output: 0

Test:2
function: fuzz.token_set_ratio
input: "Fun Stuff",""

output: 0

Test:1
function: fuzz.token_sort_ratio
input: "","testing empty string"
output: 0

Architectural Design:
fuzzyScript.sh changes directories and goes into testCaseExecutables where it executes

fuzzyWuzzyTestDriver.py will go to the testCases where it will execute each test and return whether each test past or failed

The output is then sent to log.txt by fuzzyScript.sh

Instructions:
To execute in terminal after checkout from svn

~chmod +x fuzzyScript.sh
~./fuzzyScript.sh

The output will go to log.txt

The driver may also be run separately for output in the terminal

~python fuzzyWuzzyTestDriver.py