Using Tester.py

For you assignments you have been provided a program called tester.py that you can use to check if your answers match the tests you have been given.

Step 1 Download Python 3

tester.py is a python program so if you want to run it you will need to get python 3. You can download python 3 here. You can download it here: https://www.python.org/downloads/release/python-342/. At the bottom of the page you will find the download options. Select the one that is correct for you.

Step 2 Compile Your Program

Skip this step if you are using an interpreted language like python.
Before tester.py can test you program your first have to create it. Use the appropriate compiler to create your executable, gcc for .c files, g++ for .cpp files, javac for .java files, etc. For example if the assignment asked you to create an executable called change.out, from a .c file then use gcc to create it.

Step 4 Running Tester.py

You have 2 options in how to run tester.py

1. Terminal
    1. Open terminal
    2. Change your working directory so that you are in the directory that contains tester.py. Use cd to do this
    3. On the terminal type: python3 tester.py
2. Through Idle
    1. Idle is a python editor that comes with your download of python 3
    2. Open Idle
    3. Use Idle to open tester.py
    4. Click Run - > Run Module

Regardless of how you run tester.py, tester.py, result.py, your executable, and the directories Tests and Solutions must be in one directory. If all of these files/directories are not in the same directory, then tester.py will not work.

Step 5 Interpreting the Results

After you run you will see results similar to this

```
Output: ['Please', 'enter', 'the', 'amount', 'of', 'money', 'you',
'wish', 'to', 'withdraw:', 'You', 'received', '0', 'hundred(s)',
'You', 'received', '0', 'fifty(s)', 'You', 'received', '0',
'twenty(s)', 'You', 'received', '0', 'ten(s)', 'You', 'received',
'0', 'five(s)', 'You', 'received', '0', 'one(s)']
Solution: ['Please', 'enter', 'the', 'amount', 'of', 'money', 'you',
'wish', 'to', 'withdraw:', 'You', 'received', '1', 'hundred(s)',
'You', 'received', '0', 'fifty(s)', 'You', 'received', '0',
'twenty(s)', 'You', 'received', '0', 'ten(s)', 'You', 'received',
'0', 'five(s)', 'You', 'received', '0', 'one(s)']
First mismatch occurs at word 13
Your output for that word is 0 but the correct word is 1
Test: 0-test.txt | Correct: True | Time Taken: 0.001
Test: 1-test.txt | Correct: True | Time Taken: 0.001
```

```
Test: 2-test.txt | Correct: True | Time Taken: 0.001
Test: 3-test.txt | Correct: True | Time Taken: 0.001
Test: 4-test.txt | Correct: True | Time Taken: 0.001
Test: 5-test.txt | Correct: True | Time Taken: 0.001
Test: 6-test.txt | Correct: False | Time Taken: 0.001
Test: 7-test.txt | Correct: True | Time Taken: 0.001
Test: 8-test.txt | Correct: True | Time Taken: 0.001
Test: 9-test.txt | Correct: True | Time Taken: 0.001
```

Whenever tester.py finds that your output does not match the solution it will display both your output followed by the solution. It will also tell you where it found the first mismatch in your output and what the answer was supposed to be. At the bottom tester will output a summary of what tests you passed and which ones you missed.

Important to note is that that Solution Files and tester.py's output contain only the **output** of your program and my solution. This means that you will not see the values the user input in what tester displays. If you want to see what the inputs are, you have to look in the matching test contained in the Tests folder.

Tester.py only runs the tests that it finds in the Test folder. So if you don't want to run tests 1 and 2 then you can temporarily remove them from the Tests folder and tester.py won't test them.