CS 1110, LAB 8: RECURSION EXERCISES

http://www.cs.cornell.edu/courses/cs1110/2015fa/labs/lab08.pdf

First Name: _____ **Last Name**: _____ **NetID**: _____

This lab gives you practice writing recursive functions. All of the functions in this lab are recursive functions either on sequences (e.g. strings or lists) or on integers, just as we saw in class.

This lab will have a very different format than the previous labs. We would like for you to do this entire lab in the Python Tutor. This will allow you to see *exactly* what is happening when you call a recursive function. However, if you cannot finish during lab time, then you will want to save your work. **Always remember to copy from the Tutor into a Python module before closing the web page for the Python Tutor**.

Lab Materials. We have created *a lot* Python files for this lab. This time we are not going to even try to make them available individually. Instead, you should download the ZIP file called `lab08.zip` from the Labs section of the course web page. Extract the contents of this file into a new folder on your computer.

Your folder will now contain eleven files. They will have the following names.

- `demo.py` (a module to demonstrate how to work on the lab)
- `lab08_X.py` (problem number `X` for the lab)

Getting Credit for the Lab. There are ten problems in this lab. To get credit for this lab you must **complete four of them**. You can complete any four that you wish, but we recommend that you **focus on the first four**. To get credit, show your function implementations (either in the Tutor, or saved to a Python module) to your instructor. You instructor will then swipe your ID card as usual. You are welcome to work on the remaining functions, but that is not required.

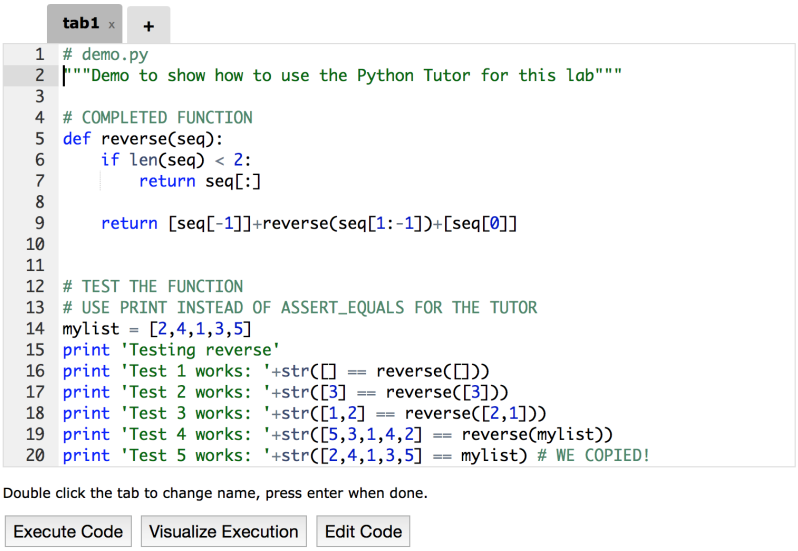
While we want you to work in the Python Tutor, always remember to save your work when done (typically in the same `lab08_X.py` file you copied into the tutor). If you close the Tutor without saving your work first, it **will be lost**.

As always, you should try to finish the lab during your section. This is a longer lab than the past two (now that you are done with all that work), and so it may not be possible. If you do not finish, you have **until the beginning of lab next week to finish it**.

1. Working with the Python Tutor

The file `demo.py` is a completed module file to show off how we plan to use the Python Tutor. Open this file and copy all of its contents into the Python Tutor. When you are done, your window should look something like the picture on the next page.

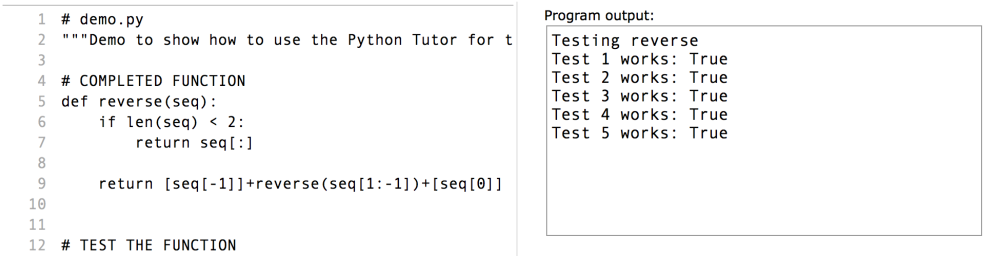
Course authors: D. Gries, L. Lee, S. Marschner, W. White



This file (as with all of the modules in this lab) has two parts: a function definition and a set of tests. Normally, we use `cornelltest` and `assert_equals` to test. However, Python Tutor does not support `cornelltest`. Instead, we are using print statements to implement our test cases.

You will note that these test cases have the same basic structure as our `cornelltest` examples. The expression on the left side of the equality is what we expect; the expression on the right is the function call. If the test succeeds, this will print out as `True`. Otherwise, it will print out as `False`.

As an example, press the button that says **Execute Code**. You will see the following:



To the right, you will see the results of the tests. The fact they are all `True` indicates that the function is working properly. If any said `False`, then that test failed.

At any time you are also free to press the `Visualize Execution` button. This is the way that we use the Python Tutor in class. It will display the folders and call frames, and allow you to step through each line of execution. If a function fails a test, we recommend that you use this feature, so that you can see what went wrong.

2. Lab Activities

There are ten files associated with this lab, all with name `lab08_X.py` (where `X` is the problem number). These files look exactly like `demo.py` except that the function body is not completed. You are complete the functions and test them in the Python Tutor. Run the tests just like we showed you with `demo.py` above. If you are having problems, visualize the execution.

You must **complete four of the ten recursive functions** provided. You can choose any four, but we recommend the first four. Once you have completed them, you are done with the lab.