**ECE 1111: Engineering Computation I**

**Laboratory No. 02: Creating Simple Python Programs**

**Goals:** The goal of this lab is to make sure you have set up the necessary infrastructure to function efficiently in this course. These assignments introduce you to the basic computing environment we will use throughout this course.

**Deposit your work in:**

/**data/courses/ece\_1111/current/labs/lab\_02/<lastname\_firstname>**

**Tasks:**

1. *Python Scripts:* Demonstrate that you can run a simple Python script that prints “hello world”. Place the source code in a file named *helloworld.py* in your lab\_02 directory and make sure the file permissions are set properly.
2. *Text Editing:* Using emacs, demonstrate that you can split your screen into two buffers. In the first buffer, you can view your hello world program. In the second buffer, you should view this file:

/data/courses/ece\_1111/current/labs/lab\_02/picone\_joseph/myprog.py

Demonstrate that you can copy and paste text from the second buffer to the first buffer. Edit the comments at the top of the file and save the result to *myprog.py* in your lab directory.

1. *Scripts vs. Programs:* Explain the difference between these two files:

/data/courses/ece\_1111/current/labs/lab\_02/picone\_joseph/myprog.py /data/courses/ece\_1111/current/labs/lab\_02/picone\_joseph/myscript.py

Copy them to your lab\_02 directory. Demonstrate how you run each. Explain the role file permissions play in the construction of these programs.

[4] *Functions:* Implement a program to compute the factorial of a number. You might find this web page useful: *https://www.geeksforgeeks.org/python-program-for-factorial-of-a-number/*. Create a file that contains a script that computes a factorial. Make sure the code conforms to our standards for formatting and commenting.

**Summary:**

Python can be thought of as both a scripting language and a programming language. In this lab, you have learned the basics of how to create Python programs and how to run them either as scripts or programs. The distinction is subtle right now, but will make more sense once we get to C/C++.