**Lab 2**

**Lab purpose**

This lab takes a step back from working with and within ArcGIS to explore Python as a language in and of itself. In this lab, you’ll be asked to solve a variety of problems using Python. Some of these problems will have immediate applied uses, while others will simply be asking you to think computationally – about what you can and cannot solve using Python and how it might be used in a variety of generalized tasks. You will also gain familiarity with the specific syntax of the Python language.

There are a wide variety of articles, guides, tutorials, and reference materials available on the Python language. You’re encouraged to read through many of these and refer to them when you run into difficulty. Make sure you understand why any solution works or you will run into significant difficulties later.

You will turn in a series of python scripts that you create to solve each problem. You will host them [in the repository here](https://github.com/UWTMGIS/TGIS501_W18/tree/master/lab2" \t "_blank). If you are using a notebook, you will enter all of your code here and upload a single .ipynb file with the format LastName.ipynb. If you are uploading individual scripts, they will be LastName\_ProblemNumber.py.

**It is strongly recommended you work through the entirety of Exercise 4 from the Scripting… book before attempting these problems. This lab may be completed using either python 2 or 3.**

## Problem 1: The Trouble With Turtles

Depending on your age, you may remember playing with a Turtle drawing program in elementary school. The history behind the Turtle is a bit longer (dating back to the 1960s) and “Turtle Graphics” generally refers to a means of drawing vector graphics on a Cartesian plane. You draw by moving an imaginary turtle around the screen. Each turtle (and there can be more than one) has a location, a pen, and an orientation. With simple commands you can draw truly complex shapes.

Follow [this link](http://openbookproject.net/thinkcs/python/english3e/hello_little_turtles.html" \t "_blank) and work up through section 3.6 (there are challenge problems below you can do for fun, but perhaps come back to them later).

### Question 1

**Write a script that asks the user to input a number and then draws a shape with that number of sides**

If you need help on getting keyboard input in python, take a look [here](http://www.python-course.eu/input.php" \t "_blank)

If you are using a notebook for this assignment, you can enter your code in the cell below, otherwise write your own script.

## Problem 2: Is GIS the best, and the looming horrors of New England

Problem 2 will have you working with text files to manipulate and analyze them. Here, we'll be working with plain text files and doing some very rudimentary forms of analysis. We'll return to these ideas later in the course using a more sophisticated approach (natural language processing), but for now we're focusing on the basics of opening files, manipulating data, and using flow control to iterate across datasets.

### Question 3

You have written a long, beautiful ode to GIS and called it GIS\_is\_the\_best.txt. Wow, you're very proud of yourself. Find this file, which you have written [in this repository](https://github.com/UWTMGIS/TGIS501_Files" \t "_blank).

You can either read it directly from the web (which we'll cover in a later lab, but you can likely figure out now if you wish) or just clone/download it somewhere local.

A couple of hints before we go one:

1. Check out the methods .upper() and .lower()
2. The assigned readings for this week covered opening files. Exercise 4 (recommended above) discusses counting words. You can also check out a quick tutorial on text files [here](http://opentechschool.github.io/python-data-intro/core/text-files.html" \t "_blank).

In the next cell (or in your own script), write a script that prints the total number of words in the document.

**That was pretty fun!**

But, don't worry, we have something even more exciting in store. It turns out that you've recently become enamored with the unspeakable horrors and non-Euclidean geometries found in H.P. Lovecraft's prose. You've decided you want to investigate his work *The Shunned House* to find two things: How many unique words he uses and how many times he uses the word "uncle".

You can find the text file for the shunned house in the same directory as your ode to GIS, [here](http://opentechschool.github.io/python-data-intro/core/text-files.html" \t "_blank).

As a hint, you make want to look into the [collections module](https://docs.python.org/2/library/collections.html" \l "collections" \t "_blank)

### Question 4

How many unique words does Lovecraft use in The Shunned House?

Like before, case does not matter; so, "whisker" and "Whisker" would be the same. Make sure you strip out punctuation, by the by, otherwise you might end up with "whisker." "whisker?" and "whisker" as separate words!

### Question 5

How many times does Lovecraft use the word "uncle" - again, case does not matter and make sure you strip punctuation.

### Bonus Questions (+1 pt each)

These questions are meant to be hard. I will only give you limited help with these. You will find some questions harder than others, you will find some questions more interesting than others. All of the questions are possible.

#### Bonus Question 1

Excluding prepositions and articles ("from", "the", "an", "with", etc.) - what are the five most frequently used words in The Shunned House?

#### Bonus Question 2

Write a script that asks for number, checks to make sure that it is in fact a number, and then finds said number’s square root (within an error of .0001). **Do not use any build in commands that find square roots (such as sqrt() or x\*\*(1/2). You must build the script using only multiplication, division, addition, and subtraction (you may also use absolute value).**

Pay attention to how many iterations it takes to solve, try to minimize it (By the way, there is a 'best' solution here, try not to look it up).

Once you lab is done - either as a series of separate scripts with names of the form LastName\_QuestionNumber.py or within this document itself (a .ipynb file), upload them to the [lab2](https://github.com/UWTMGIS/TGIS501_W18/tree/master/lab2" \t "_blank) area of the repository.