**ENGR 102**

**Lab #11 BONUS** **[100 Points]**

**Activity: Programs to test writing functions – to be done individually.**

This activity is meant to give you more experience writing functions.

In all below cases, the key idea is to write a single function. However, in all cases, **you should create a program that will essentially “test” the function**. For example, you might include several function calls and the results, or create a program where a user can enter values and see the results. You may feel free to make multiple functions within any one of these programs.

1. **[30 Points]** Imagine that you have a block of material in which a hole has been drilled:

Write a function that will take in the dimensions of the box, length, width, and height, and the radius of the hole, and determine the volume of material remaining. Assume the hole has been drilled along the height direction. Note: first write the function assuming the hole has radius less than min(length/2, width/2) – you will still receive a majority of credit (more than 70%) for this result. For full credit, you will need to account for larger radii.

1. **[10 Points]** Imagine that you have three parallel lists of the same length, one with the names of several production facilities, another with the annual cost to operate each of those facilities, and a third with the value of the products produced at each facility. Return the name and net profitability (profitability is the value of what’s produced minus the cost to operate) of the least profitable facility.
2. **[20 Points]** Write a function that takes as input a person’s name, city, state, zip code, and address, where the address is either one string (one line) or two strings (two lines), and prints the person’s information like a mailing label. Show that the routine works regardless of whether it is called with one address line or two address lines.
3. **[10 Points]** Write a function that takes the name of a file with a .csv extension (a comma-separated value file), and writes a new file that is equivalent (same name and same data) but with a .tsv extension (a tab-separated value file: like a CSV but with tabs instead of commas separating the elements of the file). Note: the character used to represent a tab is ‘\t’.
4. **[10 Points]** Write a single function that takes in a list and returns the minimum, mean, and maximum value from the list.
5. **[20 Points]** Write a function that takes in two parallel lists: a list of times (in increasing order), and a list of distance traveled by that point in time. The function should return a new list giving the average velocity between consecutive time measurements. The new list should have length one less than the original list.

Create separate functions named my\_function\_a, my\_function\_b, …. my\_function\_f. Then write the script file that calls these functions with appropriate data and demonstrates the outputs. **Generate your own data for each problem as required**. ***Zip all files required and upload one zipped file.***

Grader (TA) will test your calling script file assuming all files available in the same directory.