

CSC 280 Lab 2: Assignments and Functions

First Name: _____ last Name _____ AU ID _____

Lab is valuable time to get feedback on problem solving. Feel free to discuss with anyone in the classroom including your instructor and TA. But DO think through yourself.

The purpose of this lab is to get you comfortable with using assignment statements and functions. These are all a major part of the first assignment. The early labs are designed to get you ready for that assignment.

Getting Credit for the Lab. All of your answers should be written down on a sheet paper (or on the sheet provided to you in lab. When you are finished, you should show your written answers (Part 1) and you code and the results (for Part 2) to your instructor or TA, who will record that you did it. If you do not finish, you have until the beginning of lab next week to finish it. Remember that labs are graded on effort, not correctness.

Part 1

The first part of the lab is very similar to the previous one, in that you will be typing commands into the Python interactive shell and recording the results. At each step you should first compute the result in your head, without Python. Write the result in the second column, or “?” if you have no idea. Next type the expression or command into Python and record the result in the third column. **If the answers are different**, try to explain why in the last column.

1. Functions. Built-in functions are those that do not require you to import a module to use them. You can find a list of them in the Python documentation:

<https://docs.python.org/2/library/functions.html>

Note that the casting and typing operations are listed as functions. While this is true in Python, this is not always the case in other programming languages. That is why we treated those functions specially in the last lab.

Expression	Expected Value	Calculated Value	Reason for calculated value
min(25,4)			
max(25,4)			
min(25, max(27,4))			
abs(25)			
abs(-25)			
round(25.6)			

round(-25.6)			
round(25.64,1)			
round(26.64,0)			
round(26.64,2)			
len('truth')			

4. Using Python Module

One of the more important Python library modules is the math module. It contains essential mathematical functions like sin and cos. It also contains variables for mathematical constants such as π .

To learn more about this module, look at its online documentation:

<http://docs.python.org/library/math.html>

To use a module, you must import it. Type the following into the Python interactive shell:

```
import math
```

You can now access all of the functions and variables in math. However, to use any of them, you have to put "math." before the function or variable name. For example, to access the variable pi, you must type math.pi. Keep this in mind as you fill out that table below.

Expression	Expected Value	Calculated Value	Reason for calculated value
------------	----------------	------------------	-----------------------------

math.sqrt(9)			
math.sqrt(-9)			
math.floor(3.7)			
math.ceil(3.7)			
math.ceil(-3.7)			
math.copysign(2,-3.7)			
math.trunc(3.7)			
math.trunc(-3.7)			
math.pi			
math.cos(math.pi)			

In addition to the above expressions, type the following code into the Python interactive shell:

```
math.pi = 3
math.pi
```

What happens and Why?

Part 2

The second part of lab requires you to write small programs. Please save your script as .py file and try your best to execute it from command line/terminal in addition to directly running them from IDLE. If you still can't run command line from your machine, talk to your TA. Currently, all lab computers have default 2.7 installed so you can also run command line from the lab computers.

To consider that you complete the exercise, you **MUST show your instructor or your TA** your code and results.

1. Write a function that generates a top of a box.

```
def top():
    print ('*****')
    Then write a second function to draw the side of the box.
def side():
    print
```

Ask the user of how tall the box should be and print the box accordingly by calling the above functions. For example, the following is an example output:

```
>>>
How tall? 5
*****
*       *
*       *
*       *
*       *
*       *
*****
```

2. Write a Module (script) that includes the following arithmetic functions that does **return** the values

```
# add two numbers
def add(x,y):

# subtract two numbers
def sub(x,y):

# multiply
def multiply(x,y):

# divide
def divide(x,y):
```

```
# call the above functions to compute the value of 0.3*(23+43)/(100-1023)
```

```
Hint: add(23+43)
```

3. Imagine the owner of a movie theater who has complete freedom in setting ticket prices. The more he charges, the fewer the people who can afford tickets. In a recent experiment, the owner determined a precise relationship between the price of a ticket and the average attendance. At price \$5.00, 120 people attend a performance. Decreasing the price by a dime (\$0.1) increases attendance by 15. Unfortunately, every performance costs the owner \$180. Each attendee costs another four cents (\$0.04). The owner would like to know the exact relationship between profit and the ticket price so that he can determine the price at which he make the highest profit. The owner can enter a price (can be a float number) and return an estimated profit.

There are two ways to solve the problem. One is to extract all the math and write one function:

```
Def profit (price)
    return .... Long equation.
```

Another way is to construct a few smaller functions. For example:

```
def profit(price)
    return revenue(price)- cost(price)
```

```
def revenue(price)
```

```
    ....
```

```
def cost(price)
```

```
    .....
```

```
def attendees(price)
```

```
    return 120+ (5-price) *(15/0.1)
```