~~No starting code for this Lab. You'll need to create new~~ **~~.py~~** ~~files with the specified names given below, within your PyCharm project.~~

**~~[L2-Preamble]~~** ~~Create a new project folder~~ **~~lab\_2~~** ~~somewhere on your PC: it will hold each of the~~ **~~.py~~** ~~files you create in the steps below.~~

~~Then start PyCharm on your local PC or laptop. It should present you with a welcome window, allowing you to~~ **~~Create a New Project~~**~~, or~~ **~~Open~~**~~, or~~ **~~Check Out from Version Control~~**~~. Here, select~~ **~~Open~~**~~. If you don't see this, you might see the last project opened. In this case, select~~ **~~File->Close Project~~** ~~first, then~~ **~~Open~~** ~~within the welcome window. Navigate to your~~ **~~lab\_2~~** ~~folder, select the folder, and click on~~ **~~Open~~**~~.~~

~~You should then see the selected folder within the left~~ **~~Project~~** ~~area within PyCharm. It should be empty. Create the empty Python source file~~ **~~swap.py~~** ~~by using the command~~ **~~File->New...->Python File~~** ~~and entering the name~~ **~~swap~~** ~~(PyCharm will add the~~ **~~.py~~** ~~extension).~~

~~This new empty file will be opened within the PyCharm editing window, and you can add your code as requested below. You will likely need to configure your project with the Python interpreter installed on your PC. We'll discuss how to do this in class.~~

~~Do as many of the following as we have time for in class. Upload each of your .~~**~~py~~** ~~files to Canvas within the~~ **~~Lab 2 Assignment~~** ~~link.~~

**~~[L2-1]~~** ~~(~~**~~swap.py~~**~~) Write a program that prompts and reads two integers (~~**~~int~~**~~) from the user, assigning the first to variable~~ **~~one~~** ~~and the second to variable~~ **~~two~~**~~. Then write code which exchanges or swaps the values in~~ **~~one~~** ~~and~~ **~~two~~**~~.~~

**~~[L2-2]~~** ~~(~~**~~eval.py~~**~~) This is practice for~~ **~~H1-2~~**~~. Write a program which does the evaluation of the expression:~~ **~~10 + 8 \* 6 - 4 / 2 - 9~~** ~~via a series of assignment statements. Each assignment statement must be of the form~~ ***~~var = e1 op e2~~***~~, where~~ ***~~var~~*** ~~is some variable,~~ ***~~op~~*** ~~is one of the arithmetic operators~~ **~~+,-,/,\*~~** ~~and each of~~ ***~~e1~~*** ~~and~~ ***~~e2~~*** ~~should either (a) an~~ **~~int~~** ~~literal or (b) a variable you assigned to in earlier statements. Your final assignment statement should be of the form~~ **~~result~~** ~~=~~ ***~~e1 op e2.~~***

**~~Note that each of your assignment statements should have only ONE operator and TWO operands on the right-hand side.~~**

~~After this, add the two print statements~~ **~~print(result)~~** ~~and~~ **~~print(10+8\*6-4/2-9).~~** ~~Be sure you evaluate the sub-expressions in the correct order when calculating~~ **~~result~~**~~, so that the output of both is the exactly the same. Why isn't your answer an~~ **~~int~~**~~? And don't forget that~~ **~~47~~** ~~is NOT the same as~~ **~~47.0~~**~~!~~

**~~[L2-3]~~** ~~(~~**~~noexp.py~~**~~) Write a program which reads integers~~ **~~x~~** ~~and~~ **~~n~~**~~, then prints out the value of~~ **~~x~~~~n~~**~~; that is,~~ **~~x~~** ~~raised to the~~ **~~n~~**~~th power. HOWEVER: you cannot use the~~ **~~\*\*~~** ~~operator. Hint: use the accumulator pattern... Compute~~ **~~r = x \*\* n~~** ~~in the following way:~~

**~~result = 1~~**

**~~for counter in range (n):~~**

**~~result = result \* x~~****~~# result now contains x\*\*n~~**



~~~~**~~[L2-4]~~** ~~(~~**~~boxes1.py~~**~~) Create a Python program using turtle graphics (~~**~~import turtle~~**~~) that draws the picture shown at the right. One square is nested in the other, with the inner and outer squares as 80 and 100 units on a side, respectively. The horizontal and vertical distance between squares is 10 units, and both squares are centered at the turtle world's origin of (0,0). If you have trouble getting your drawing to look exactly like the picture, do the best you can.~~

**~~[L2-5]~~** ~~(~~**~~triangles.py~~**~~) Create a Python program using turtle graphics that draws the picture shown at the right. The inner equilateral triangle is nested inside the outer equilateral triangle, with the outer having a side of length 200. Each vertex of the inner triangle touches the midpoints of sides of the outer triangle. You can draw your triangles anywhere within the screen; for an extra challenge, see if you can center both triangles at the turtle's origin~~ **~~(0,0)~~**~~. This means the origin must be the same distance from each vertex of~~

~~the inner triangle.~~

**~~[L2-6]~~** ~~(~~**~~rangefun1.py~~**~~) Write a program that reads integers~~ **~~a,b,c~~** ~~from the user using a sequence of three~~ **~~input()~~** ~~statements (don't forget to cast the string input to int!), then uses the following loop to print out each element of the~~ **~~range(a,b,c)~~**~~, one per line:~~**~~for num in range(a,b,c):~~**

**~~print(num)~~**

~~Run your program multiple times, entering different values for~~ **~~a~~**~~,~~ **~~b~~**~~, and~~ **~~c~~**~~. Be sure you understand how~~ **~~range(a,b,c)~~** ~~generates the different sequences that are printed out, especially for negative~~ **~~c.~~**