CS160 Computer Science I
Lab 7

Objectives
Work with functions
Work with graphics

Part 1

Assignment
Write each of the following functions using Python. The function header MUST be written as specified. In your main code test all of the specified functions. Each function must have a comment block explaining what it does, what the parameters are and what the return value is.

Please remember the following two guidelines:
- unless the purpose of the function is to generate output DO NOT write to the screen within the function (output for debugging purposes does not apply)
- unless the purpose of the function is to ask the user for a value DO NOT ask the user for a value within the function

Required functions
`def square (intValue):`
This function returns the square of `intValue.`

`def summation (intValue):`
This function returns the summation of 1 to `intValue`. You can assume that `intValue` will be positive. For example, `summation (5)` would return 15 (1 + 2 + 3 + 4 + 5).

`def sumOfSquare (intValue):`
This function return the sum of the squares from 1 to `intValue`. For example, `sumOfSquares(5)` would return 55 (1 + 4 + 9 + 16 + 25). You **MUST** use a loop and the `square` function to determine the returned value.

`def factorial (intValue):`
This function returns the factorial of 1 to `intValue`. You can assume that `intValue` will be positive. For example, `factorial (5)` would return 120 (1 * 2 * 3 * 4 * 5).

`def distance (x1, y1, x2, y2):`
This function returns the distance between two points. x1, y1 will be the first point, and x2, y2 will be the second point. The formula is `distance` $= \sqrt{(x2-x1)^2 + (y2-y1)^2}$. You will want to look at the functions offered by the `Math` class to help with this formula.

`def isOdd (int intValue):`
This function will return `True` if `intValue` is odd, otherwise it returns `False`.

`def isEven (intValue):`
This function will return `True` if `intValue` is even, otherwise it return `False`. This function **MUST** use `isOdd` to determine the returned value. The point behind this function is to minimize the amount of redundant work that is done in our code, not that determining odd or even is a lot of work. It's the concept more than the reality of the code in this case.

Part 2

Assignment
Write a program using turtle graphics which writes your initials, or any other three unique letters, to the display. Look to your program for lab 1 for reminders of how to use turtle graphics.

Functions that must be written
def drawLetter (x, y, letterColor):
Write three functions, one for three unique letters. Personally, I would write drawT, drawL, and drawS. Each function must draw that letter to the screen. The x and y values determine the upper left-hand location for the start of the letter. Ensure that each letter takes the same amount of space, both vertically and horizontally. The letterColor parameter will be a string used to determine the color of the letter – use it to set the fillcolor, and maybe pencolor. You can also set pencolor to "black" to create an outline of the letter. In the Turtle Graphics Resources link in blackboard I have a link to valid turttle graphic colors.

Each function can have **one**, and only one, goto command. If drawing a letter which requires two "shapes," such as a "O" with a two circles you can use a second goto. The initial goto should position the turtle in the upper left-hand location of the letter to be drawn. It might be smart to ensure that you know what direction the turtle is pointing before starting to draw the letter. All of the actual drawing must be done with basic movements, such as forward, backward, left, or right. Remember to use penup and pendown to ensure you only draw the desired lines.

def drawShadowedLetter (x, y, letterColor, offset):
Write three more functions, one for each letter (mine would be drawshadowedT, drawShadowedL, and drawShadowedS). These functions will use your drawLetter functions, once using letterColor, and once adding offset to the x and y values and using "black" as the color. This will create a "shadowed" letter. These three functions **will not** be doing any actual drawing, all of the graphics work will be done in the drawLetter functions.

In the main program, draw your initials to the screen twice. The first time write your initials to the screen using the drawLetter functions. Somewhere else on the display, write your initials to the screen using the drawShadowedLetter functions.

Challenge – no points, just knowledge ☺
Write a function named drawName (x, y), which writes all the letters to the screen, evenly spaced out. This will require a loop to access each letter in your name, as well as a variable to determine the x, y location for each letter. You could even write drawHName(x, y) and drawVName (x, y), which would write out your letters either horizontally or vertically, again evenly spaced, using x and y as a starting location for the first letter.

Requirements
Complete comment section that includes your name, id number, program number and a brief description of the prog