**CIS 229 – Python Programming – Programming Assignment**

**Shape Drawing**

Overview

In this assignment, the student will create a Python script that implements a series of functions that draw scaling shapes on the output console.

When completing this assignment, the student should demonstrate mastery of the following concepts:

* Functions
* Loop Syntax (While Loops, For Loops)
* Lists Used As Iterators
* Variable Scope
* Function Inputs
* Algorithm Design – Nested Loops

Assignment

In this assignment, you will be using loops to draw a variety of geometric shapes on the console screen. In all cases, the shapes should be drawn with asterisk (\*) symbols. While writing the functions keep in mind that when the print() function in Python is used, a newline will automatically be generated. You will most likely need to suppress this behavior for drawing. This can be accomplished by using the “end” key word argument (kwarg) built in to the print() function.

Example

*print(‘This displays with a newline.’)*

*# this will cause a newline to be displayed after the*

*# period at the end of the sentence*

*print(‘This displays without a newline.’, end = “”)*

*# this will suppress the newline behavior. Anything added*

*# between the double quotes will print at the end*

*# of the line*

Although the end key word argument can be used to suppress the new line, anything can be passed to the end kwarg to display at the end of the line. Spaces can be used to help re-create the type of drawings provided in the examples below (notice that each asterisk in the drawings is followed by a single space). The space also helps to give the shapes a proportional appearance.

The functions are described below:

Function #1 – Drawing a Rectangle

This function should take two arguments. The first one contains the width of the rectangle and the second one contains the height of the rectangle (in asterisks). When writing this function, only use ‘for’ loops to create the display.

Function #2 – Drawing a Triangle

This function should take a single argument for the height of the triangle in asterisks. The display should produce a right isosceles triangle with the right angle appearing at the bottom-left corner of this display. When writing this function, only use ‘while’ loops to create the display.

Function #3 – Drawing a Pyramid

This function should take a single argument representing the base radius of the pyramid. The base radius represents half the length of the base of the pyramid. You should include a single column of asterisks going down the center of the pyramid that do not count towards the base radius. Consider the example below:

 \*

 \* \* \*

 \* \* \* \* \*

 \* \* \* \* \* \* \*

The base radius of this triangle is 3. That is because there are 3 asterisks to the left and right of the center column of the Triangle. If we replace the center column with C’s, this becomes apparent.

 C

 \* C \*

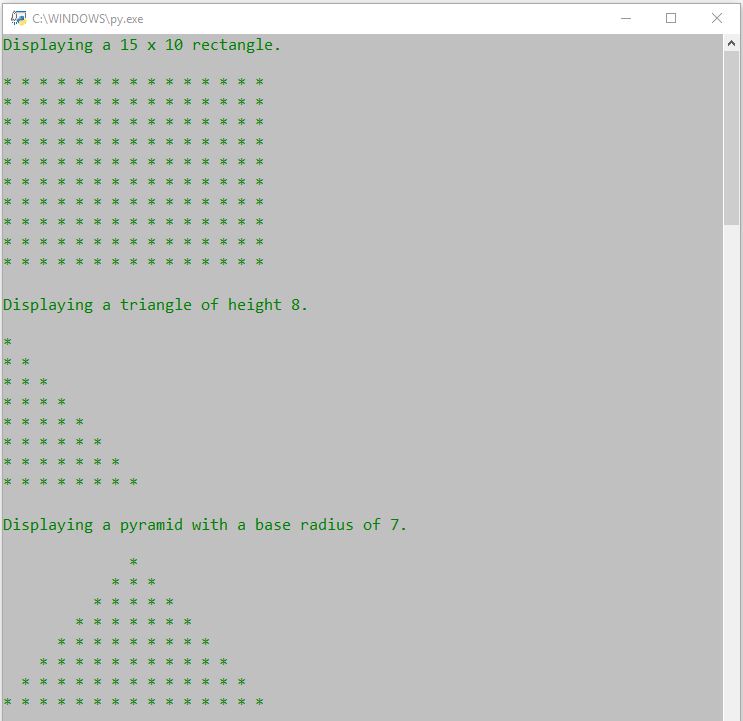
 \* \* C \* \*

 \* \* \* C \* \* \* <= notice 3 asterisks to the left and right of the bottom of the pyramid

Please keep in mind that although the pyramid appears to be a triangle, the logic needed to properly draw it is more complicated than the logic used to draw the triangle in the previous function. You will need to develop some sort of technique to pad the left and right side of each row in the pyramid’s display to create the centering effect. When writing this function, you can use any type of loop that you wish.

For the sake of simplicity, you can hard-code the values that are passed into each function. However, the logic of each function must operate in a completely abstract manner and handle any realistic input values that would generate a shape that fits on the standard console screen.

Example Screen Shot



Assessment

This assignment will be assessed based on the provided grading rubric.