

CSE 8A Programming Assignment 2

Fall 2019

Due Date: Mon. Oct 14th, 11:59 PM (PDT)

Learning Goals:

- Use both a `for` loop and a `while` loop to control the python Turtle
- Call a function that takes a parameter from inside another function and inside a loop
- Get creative and have fun getting the computer to draw all sorts of shapes and colors

Submission:

- The submission template can be found [here](#).

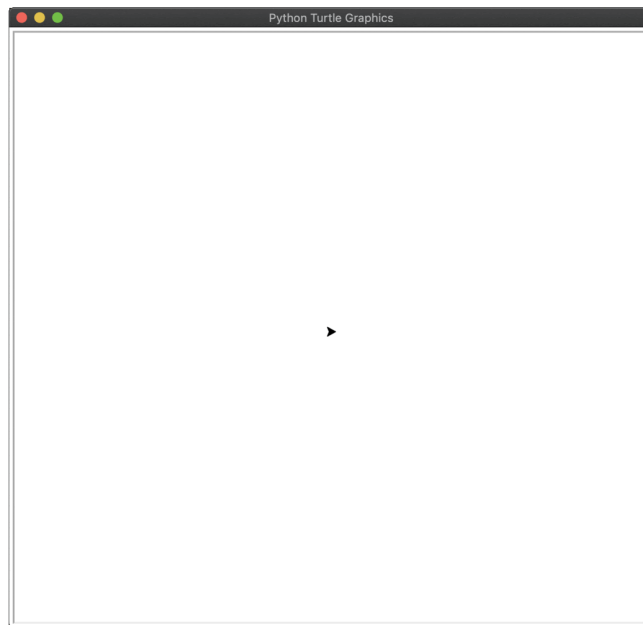
Part 0: Warm Up (Optional)

In this assignment you will use python's built-in [Turtle graphics](#) package to draw creative shapes on the screen. This package contains many functions that you can use to control a "turtle" by making it move around the screen leaving a path behind which you'll use to draw shapes. To get started, simply launch python3 and execute `>>> import turtle`.

Example: Drawing a square

Open idle3 and try the following instructions for yourself.

```
>>> import turtle
>>> turtle.showturtle()
>>> turtle.forward(100)
>>> turtle.right(90)
>>> turtle.forward(100)
>>> turtle.right(90)
>>> turtle.forward(100)
>>> turtle.right(90)
>>> turtle.forward(100)
>>> turtle.hideturtle()
```



Reading Documentation

While the course staff is available to answer your questions on Piazza and Lab/Office Hours, know that you also have the package documentation at your disposal. For instance, say you want to know more about what the `turtle.penUp()` function does. You can do so by reading its documentation [here](#).

Adapted from

http://www.cse.msu.edu/~cse231/PracticeOfComputingUsingPython/02_Control/Draw/Project02.pdf

Part 1: Turtle Art

In this part of the assignment, you will draw a picture containing multiple shapes of multiple colors and arranged to be visually pleasing. Although you are free to decide the precise shape(s) and layout for your picture, your program must adhere to the following requirements:

- The program must contain at least one function, named `drawPart`, that you have defined that takes at least one numerical parameter. `drawPart` must draw some part of the overall drawing, and something about the part drawn must depend on the parameter's value. For example, the parameter might specify the size of the part, the number of sides, the angle, etc.
- The program must use at least one `while` loop and at least one `for` loop to create the drawing. One (or more) loops can be inside the `drawPart` function you define, but at least one must be outside.
- The function `drawPart` must be called from inside a loop in the program.
NOTE: You should **NOT** call the function `drawPart` from within its own body (itself)!
- The drawing produced by the program must depend on at least one input from the user. This input might determine the number of subshapes, the size of the smallest shape, the starting location of the drawing, etc. It's up to you.
NOTE: Provide a meaningful prompt to the user when asking for `input()`

You may make your drawing as simple or as elaborate as you'd like, but it must be your own design/creation. While you can refer to examples from class and from the assigned resources, you may **NOT** copy code from the web in creating your program.

Part 2: Test Cases

Include the following test cases and outputs with your submission:

- The output of three different calls to your function `drawPart` with three different arguments. Include the line of code that performs the call and the resulting visual output. Make sure these tests illustrate how changing the argument changes what is drawn.
- The output of running your full program three different times with three different inputs from the user. Make sure these tests illustrate how changing the user's input changes the drawing.

In all cases, explain why these test cases were chosen and why they illustrate the scope of your program's functionality.

Include in your report any known bugs or issues with your program. If you think you have none, justify why.

Part 3: Reflection (individual)

Once you have finished and submitted your assignment, fill out the reflection form [here](#). Don't forget that EACH STUDENT must fill out their own reflection to get credit.