Cpts 111 Lab #4

In this lab, you will use the following constructs to solve 3 tasks:

1. IF / ELSE statements [tasks 1, 2, & 3]
2. Turtle Graphics [tasks 2 and 3]

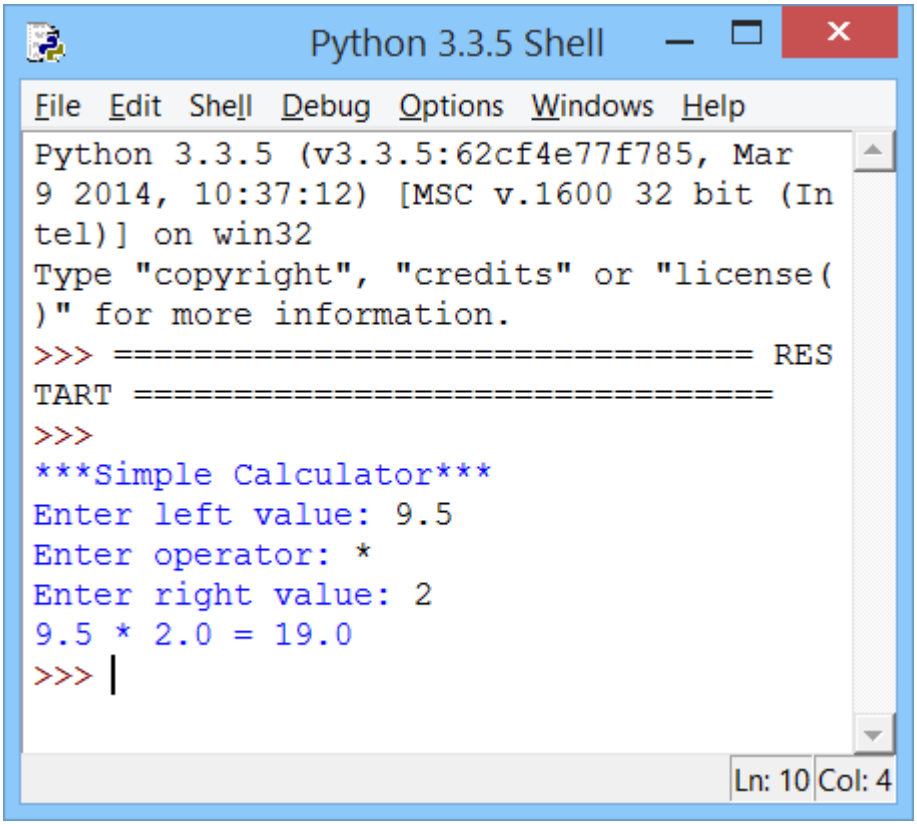
In addition, this lab utilizes the following constructs which are provided for you:

1. WHILE loops

Task 1

In this lab, you will write a simple calculator that can perform addition, subtraction, multiplication, and division. The program flow for your calculator is as follows:

1. Prompt user for left-hand value
2. Prompt user for operator (e.g. +,-,*,/)
3. Prompt user for right-hand value
4. Output result

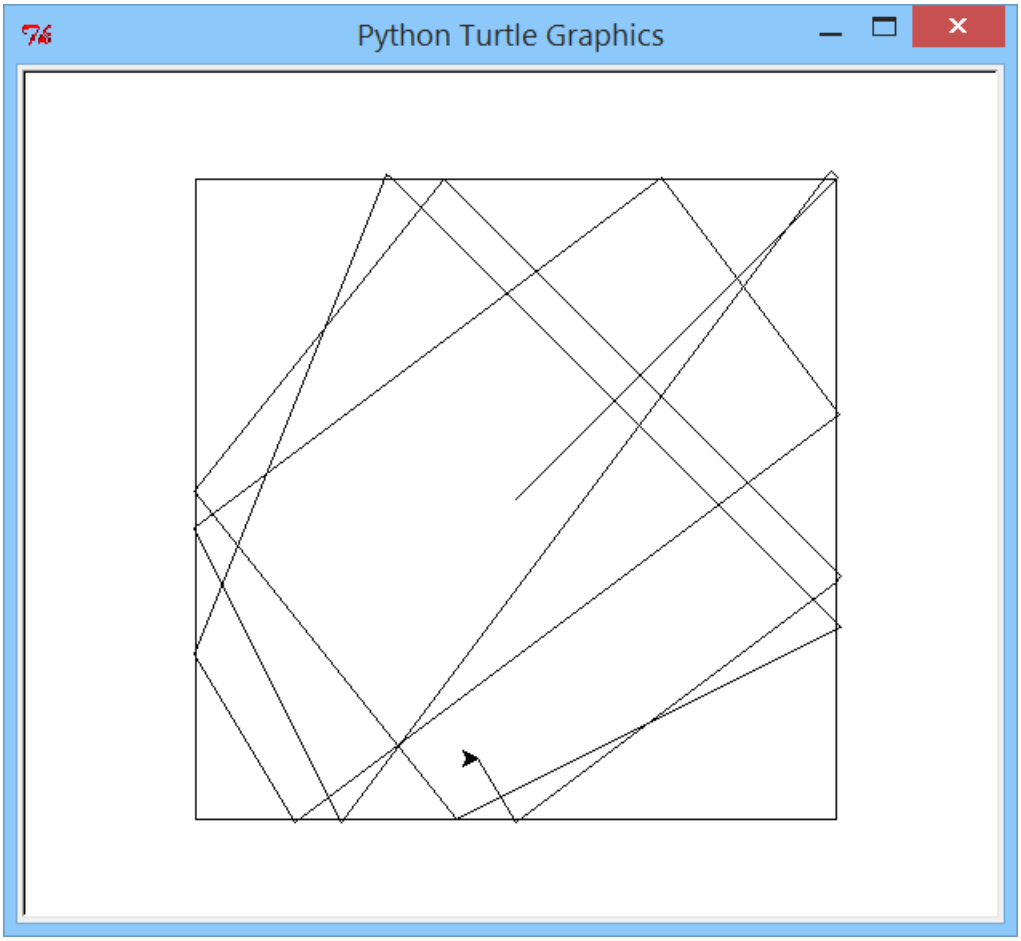


Task 2: Keeping a turtle in-bounds part 1

In this task, you must modify the supplied starter code to keep the turtle within the drawn box. Search the starter code for the "LAB 3 TODO" comment and make your changes there. Note that you can use the x_position and y_position variables to check to see if the turtle is within the box's boundary (-200 to 200). If the turtle moves outside the boundary, you will need to change one or more of the following variables:

- y_speed
- x_speed
- move_x_positive
- move_y_positive

Here's an example of my turtle that is kept in bounds:



Task 3: Keeping a turtle in-bounds part 2

Like the previous task, your goal is to keep the turtle within the bounding box (-200 to 200). However, this time, you can only use the left() and right() turtle rotation functions to alter the turtle's behavior. Again, search for the "LAB 3 TODO" comment in the supplied task 3 starter code. In order to determine if the turtle has gone out of bounds, you will need to use one or more IF statements based on the variables "turtle_x" and "turtle_y", which track the turtle's x and y position. Here's a screenshot of my working program: