

Python Turtle Module Programming

In this programming exercise we were introduced to the *Turtle* module within python and learned to use functions to execute desired outcomes.

Laboratory Exercise:

Experiment with the Turtle Module

Source Code:

```
import turtle

from turtle import(clone)

turtle.bgcolor("black") #Set Screen Color

Marius=turtle.Turtle() #Set Turtle Name

turtle.showturtle() #Display Turtle

Marius.shape("turtle") #Set Turtle Shape

Marius.pencolor("orange")#Set Pen Color

Marius.fillcolor("green") #Set Turtle Color

Marius.shapesize(3,3,3)

Marius.pen(5) #Set Pen Length

Marius.forward(100) #Moves Turtle Right 100 Pixels

Marius.left(90) #Move Turtle Left 90 Degrees

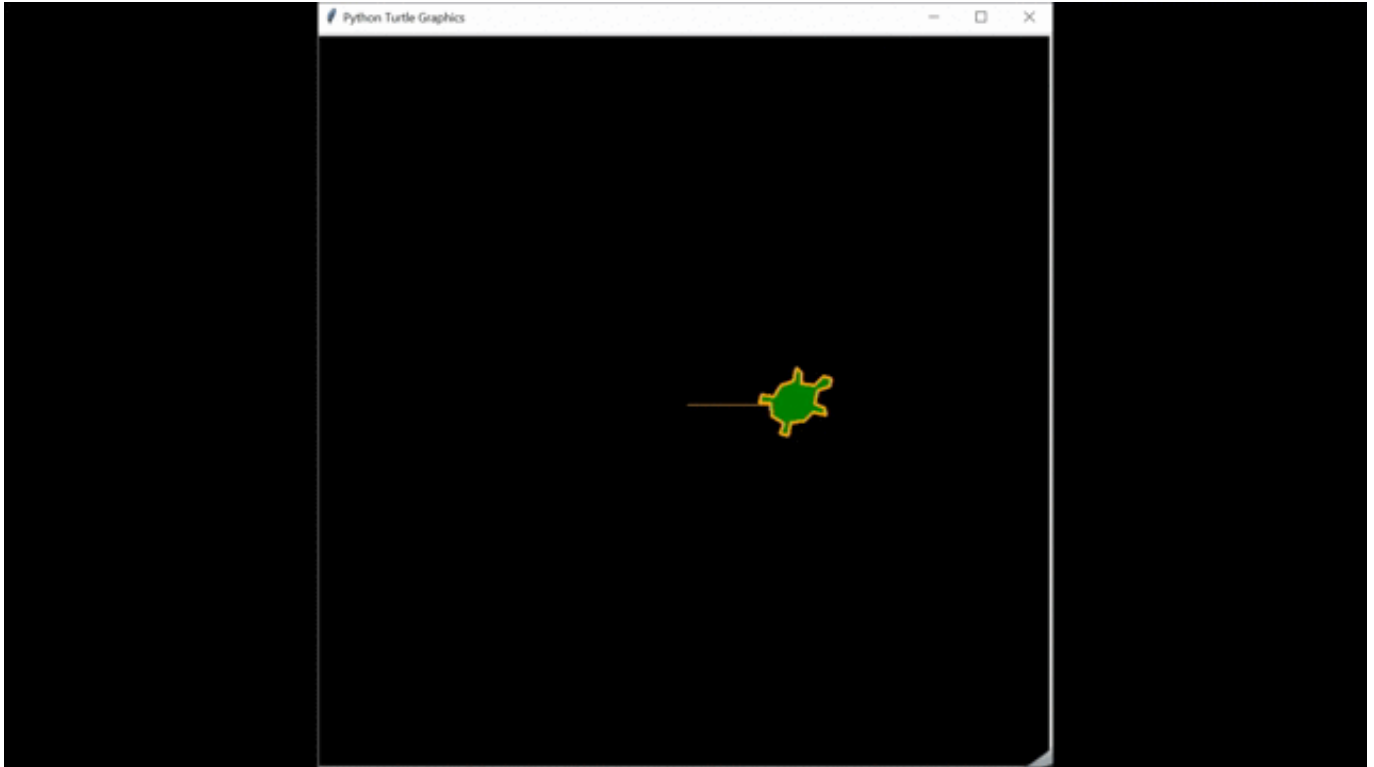
#For Loop to Make a Square

for x in range (4):

Marius.forward(100)
```

Marius.left(90)

The following is a visual of the program's output:



Click to View Media

Programming Assignment:

Create the following turtle graphic

The following lab should be uploaded on your ePortfolio when is finished.

Using the turtle and random modules from Python, write a program (or script) to create an output screen as the one shown in the figure below.

To achieve this output you have to do the following:

- Change the color of the window to "red" using wn.bgcolor property of the screen variable you create.
- Your turtle variable has to have a pen color "white", you have to use the .pencolor property of your turtle variable.
- You have to make your turtle to change its heading to the right with an random angle between 0 and 45 degrees.

- Move your turtle forward a random distance between 0 and 150 pixels.
- Then move the turtle backward the same distance than above
- Repeat the last three steps 100 times

Source Code:

```
import turtle

import random

turtle.bgcolor("red")

turtle.showturtle()

Galib=turtle.Turtle()

Galib.color("white")

Galib.shape("turtle")

Galib.pen(5)

for i in range (100):

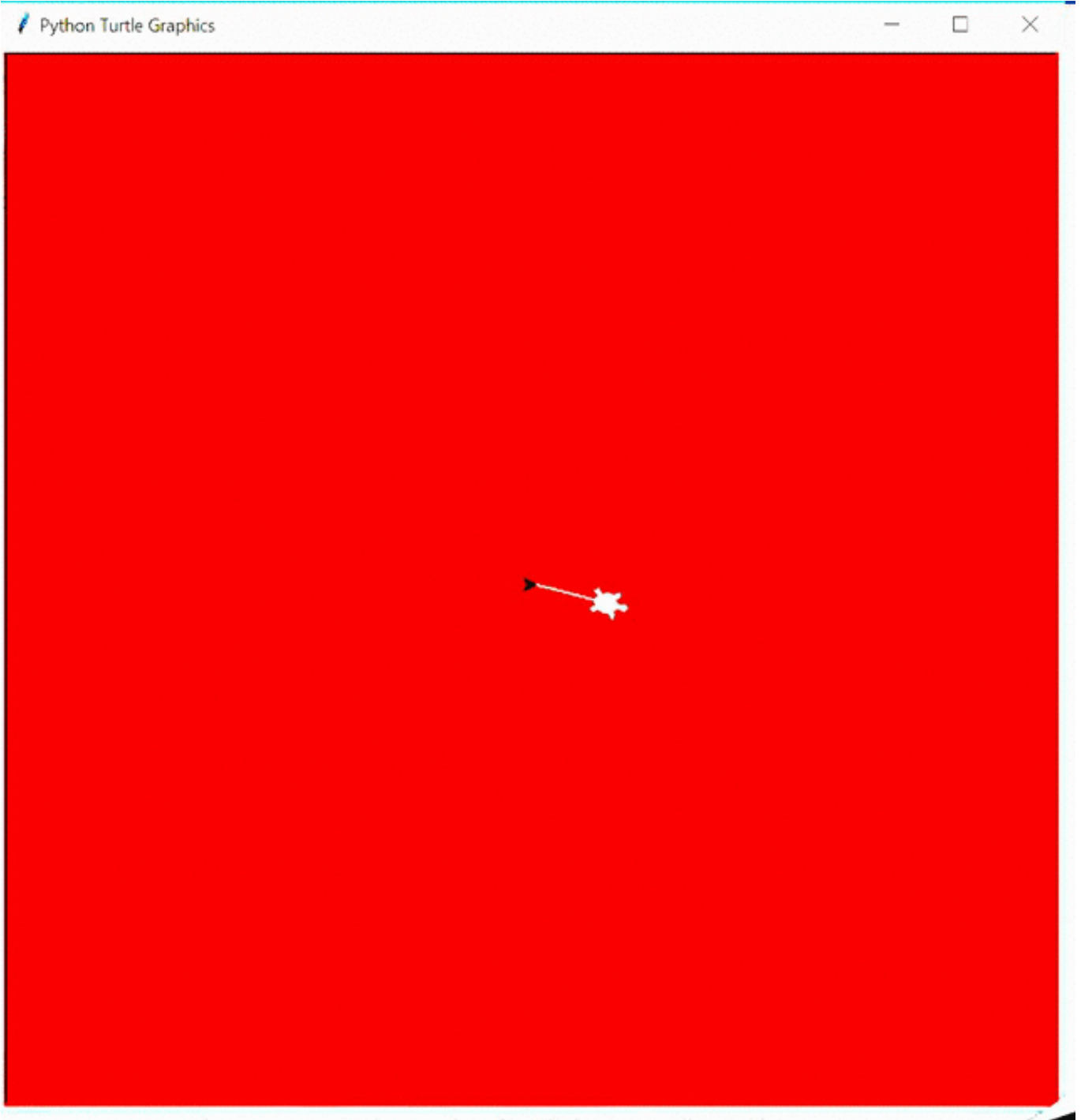
    Galib.right(random.randint(0,45))

    dist_Move=random.randint(0,150)

    Galib.forward(dist_Move)

    Galib.backward(dist_Move)
```

The following is a visual of the program's output:



[Click to View Media](#)