**Name: Session:**

**Programming I**

**Lab Exercise 9.1.22**

**Getting Started with Python Turtles**

Here is a short little program to introduce you to the power of Python. It uses the Turtle module to draw some cool graphics.

Before you do anything, create a folder called MyPython on your desktop. You will use this folder to store all of your programs.

1. Start IDLE (the Python source code editor). This will start the Python Shell (provide a facility to get input to and output from your program)
2. From the File menu, select New Window. This will open a new editor window.
3. Type in the following code and save it as hexSpiral.py. This code is **case sensitive** and indentation is **absolutely critical**. Also notice that the first line that starts with a hashtag is a comment (the name of program in this case).

**#HexSpiral.py**

import turtle

colors=['red', 'purple', 'blue', 'green', 'yellow', 'orange']

window = turtle.Screen()

t=turtle.Pen()

turtle.bgcolor('black')

for x in range(360):

t.pencolor(colors[x%6])

t.width(x/100+1)

t.forward(x)

t.left(59)

window.exitonclick()

1. You can run this program by either hitting the F5 key or select Run Module from the Run menu.

To help you understand what this code does, we will type in and run a series of short turtle programs.

**# SquareSpiral1.py**

import turtle

window = turtle.Screen()

t=turtle.Pen()

for x in range(1,100):

t.forward(x)

t.left(90)

window.exitonclick()

**#SquareSpiral2.py**

import turtle

t=turtle.Pen()

window = turtle.Screen()

for x in range(100):

t.forward(x)

t.left(91)

window.exitonclick()

**#CircleSpiral1.py**

import turtle

window = turtle.Screen()

t=turtle.Pen()

for x in range(100):

t.circle(x)

t.left(91)

window.exitonclick()

**#SquareSpiral3.py**

import turtle

window = turtle.Screen()

t = turtle.Pen()

t.pencolor("red")

for x in range(100):

t.forward(x)

t.left(91)

window.exitonclick()

**#ColorSquareSpiral.py**

import turtle

window = turtle.Screen()

t = turtle.Pen()

colors = ["red", "yellow", "blue", "green"]

for x in range(100):

t.pencolor(colors[x%4])

t.forward(x)

t.left(91)

window.exitonclick()

**#ColorCircleSpiral.py**

import turtle

window = turtle.Screen()

t = turtle.Pen()

turtle.bgcolor("black")

colors = ["red", "yellow", "blue", "green"]

for x in range(100):

t.pencolor(colors[x%4])

t.circle(x)

t.left(91)

window.exitonclick()

**# ColorSpiral.py**

import turtle

window = turtle.Screen()

t = turtle.Pen()

turtle.bgcolor("black")

# You can choose between 2 and 6 sides for some cool shapes!

sides = 6

colors = ["red", "yellow", "blue", "orange", "green", "purple"]

for x in range(360):

t.pencolor(colors[x%sides])

t.forward(x \* 3/sides + x)

t.left(360/sides + 1)

t.width(x\*sides/200)

window.exitonclick()

**Programming Challenges**

**#1: CHANGING THE NUMBER OF SIDES**

We used a variable, sides, in the *ColorSpiral.py* program, but we didn’t vary it much or change its value except for editing, saving, and running the program again. Try changing the value of sides to another number, say 5. Save and run the program to see how this affects your drawing. Now try 4, 3, 2, and even 1!

Now, add two or more colors, in quotes, separated by commas, to the list of colors in the sixth line of the program.

Increase the value of sides to use this new number of colors — try 8 or 10 or more!

**# ColorSpiral10.py**

import turtle

window = turtle.Screen()

t=turtle.Pen()

turtle.bgcolor('black')

# You can change sides between 2 and 10 for some cool shapes!

sides=10

colors=['red', 'yellow', 'blue', 'orange', 'green', 'purple', 'gray', 'white', 'pink', 'light blue']

for x in range(360):

t.pencolor(colors[x%sides])

t.forward(x \* 3/sides + x)

t.left(360/sides + 1)

t.width(x\*sides/200)

window.exitonclick()

**#2: HOW MANY SIDES?**

What if you want to let a user decide the number of sides while the program runs?

sides = input("Enter a number of sides between 2 and 6: ")

Replace the line sides = 6 in *ColorSpiral.py* with the preceding line. Your new program will ask how many sides the user wants to see. Then, the program will draw the shape the user asks for. Give it a try!

**# ColorSpiral\_Challenge.py**

import turtle

window = turtle.Screen()

t=turtle.Pen()

turtle.bgcolor('black')

# You can change sides between 2 and 6 for some cool shapes!

sides=input('Enter a number of sides between 2 and 6: ')

colors=['red', 'yellow', 'blue', 'orange', 'green', 'purple']

for x in range(360):

t.pencolor(colors[x%sides])

t.forward(x \* 3/sides + x)

t.left(360/sides + 1)

t.width(x\*sides/200)

window.exitonclick()

**#3: RUBBER-BAND BALL**

Try changing the *ColorSpiral.py* program into a more tangled and abstract shape just by adding an extra turn inside the end of the drawing loop. Add a line like t.left(90) to the bottom of the for loop to make the angles sharper (remember to indent, or space over, to keep the statement in the loop). The result, looks like a geometric toy or perhaps a ball made of colored rubber bands.

**# RubberBandBall.py**

import turtle

window = turtle.Screen()

t=turtle.Pen()

turtle.bgcolor('black')

sides=6

colors=['red', 'yellow', 'blue', 'orange', 'green', 'purple']

for x in range(360):

t.pencolor(colors[x%sides])

t.forward(x \* 3/sides + x)

t.left(360/sides + 1)

t.width(x\*sides/200)

t.left(90)

window.exitonclick()