**Outline**

t.b.d.

**Objectives**

* tbd

**Materials**

* tbd

**Level 0: Teacher Demo of Sample Programs**

1. Sample program #1 is an example of a "Syntax Error". Follow the teacher demo and explain the characteristics of a syntax error. Consider the following criteria:  
   1. Did the program have an error before starting to run?  
      Yes it did.
   2. Did the program encounter an error before it finished running?  
      Yes because there were two errors that needed to be fixed.
   3. Did the program do what it was supposed to do?

After fixing the errors, the three circles were displayed on the screen.

1. Sample program #2 is an example of a "Run-time Error". Follow the teacher demo and explain the characteristics of a run-time error. Consider the following criteria:  
   1. Did the program have an error before starting to run?  
      Yes it did.
   2. Did the program encounter an error before it finished running?  
      Yes it did because the program only displayed two circles when the pixels rgb for three circles was given.
   3. Did the program do what it was supposed to do?

After changing the range from 4 to 3 and changing the circleNumber to 1 the program showcased three circles on the screen.

1. Sample program #3 is an example of a "Logic Error". Follow the teacher demo and explain the characteristics of a logic error. Consider the following criteria:  
   1. Did the program have an error before starting to run?  
      Yes it had three errors.
   2. Did the program encounter an error before it finished running?  
      Yes because the colours for the pixels were provided but the circles were coming as black circles.
   3. Did the program do what it was supposed to do?

After adding myPen.color(rgb) and changing the range from 2 to 3 and changing the num0fCircles to 0 the program was executed and displayed the coloured circles.

**Level 1: Syntax Errors**

1. Research the definition of the word "Syntax". Summarize its meaning below and how it relates to computer languages and programming.

An arrangement of words and phrases that create well-formed sentences. In computer language and programming it is a set of rules that defines the combinations of symbols that are considered to be a correctly structured document or sentence.

1. Research the definition of a "Syntax Error" related to computer programming. Summarize this definition below.

A syntax error is an error in the source code of a program.

OR

A syntax error is an error in the syntax of a sequence of characters or tokens that is intended to be written in a particular programming language.

1. Explain why Sample Program #1 is an example of a "Syntax Error".

In line , after myPen.down there was only one bracket.

myPen.down(

There has to be () both brackets.

Also the spelling in line 17 was circleColours but the program said that circleColours was not specified so I went to line 4 and changed the spelling from circleColors to circleColours.

This is an example of syntax error because there was an error or should I say errors in the source program.

1. Find and correct the syntax errors in Sample Program #1. Provide a listing of your corrected program below.
   * Use a "#" at the beginning of each line containing an error   
     to "Comment Out" the bad code
   * List the corrected code line underneath the commented out error line

import turtle

myPen = turtle.Turtle()

# circleColors = [(196,196,0),(196,0,196),(0,196,196)]

circleColours = [(196,196,0), (196,0,196), (0,196,196)]

def drawCircle(rgb) :

# myPen.down(

mypen.down()

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 0

for circleIndex in range(3) :

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

**Level 2: Run-time Errors**

1. Research the definition of a "Run-time Error" related to computer programming. Summarize this definition below.

A run-time error is a program that occurs while the program is running.

1. Explain why Sample Program #2 is an example of a "Run-time Error".

This is an example of a run-time error because the program continues but stops at the last circle during run-time because the pixels of the fourth circle are not provided.

1. Find and correct the run-time errors in Sample Program #2. Provide a listing of your corrected program below.
   * Use a "#" at the beginning of each line containing an error   
     to "Comment Out" the bad code
   * List the corrected code line underneath the commented out error line

import turtle

myPen = turtle.Turtle()

# circleColours = [(196,196,0),(196,0,196),(0,196,196)]

circleColours = [(196,196,0),(196,0,196),(0,196,196)(196,196,0)]

def drawCircle(rgb) :

myPen.down()

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

# circleNumber = 1

CircleNumber = 0

for circleIndex in range(4) :

print(circleNumber)

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

1. Explain the difference between a "syntax error" and a "run-time error".

A syntax error is an error that happens while writing a program whereas a run-time error happens while executing a program.

**Level 3: Logic Errors**

1. Research the definition of a "Logic Error" related to computer programming. Summarize this definition below.

Logic error occurs when a program does not to what the programmer expected it to do.

1. Explain why Sample Program #3 is an example of a "Logic Error".

This is an example of logic error because the program was intended to be made for three circle but the range was written as 2. This was an error made by the programmer. The programmer also forgot to put myPen.color(rgb) and because of that the circles were coming black and not coloured.

1. Find and correct the logic errors in Sample Program #3. Provide a listing of your corrected program below.
   * Use a "#" at the beginning of each line containing an error   
     to "Comment Out" the bad code
   * List the corrected code line underneath the commented out error line

import turtle

myPen = turtle.Turtle()

circleColours = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down()

# myPen.color(rgb) – This was missing

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

#numOfCircles = 2

numOfCircles = 3

for circleIndex in range(2) :

for circleIndex in range (3)

# circleNumber = numOfCircles – circleIndex

circleNumber = numOfCircles – circleIndex - 1

# print(circleNumber)

This part will be deleted

drawCircle(circleColours[circleNumber])

1. Explain the difference between a "logic error" and a "syntax error".

A syntax error is an error made while writing the program meaning the program doesn’t understand the grammar of a programming language. Logic error is when the program doesn’t do what the programmer expects it to do.

1. Explain the difference between a "logic error" and a "run-time error".

Logic error produces the wrong input and a run-time error is a program error that occurs while the program is running.

**Level 4: Your Sample Program**

1. Create a sample program to show the different types of programming errors. Provide your program listing below.

* Your program must be of your own design and must be different from the sample programs provided in this module.
* Your program must contain at least one example of each of: a syntax error, a run-time error, and a logic error.
* Provide the corrected code in a comment underneath the error code (using a "#" at the beginning of the comment line).

**import turtle**

**myPen=turtle.Turtle()**

**posX=0**

**posY=0**

**pixelAddress= 0**

**pixelMemory = [**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,255,0),(0,255,0),(0,255,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),**

**(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0),(0,0,0),(0,255,0),(0,0,0),(0,0,0)**

**]**

**def drawPixel(rgb) :**

**global posX**

**myPen.down()**

**myPen.color(rgb)**

**myPen.begin\_fill()**

**myPen.circle(8)**

**myPen.end\_fill()**

**myPen.up()**

**myPen.forward(18)**

**posX = posX + 18**

**def newRow( :**

**#def newRow():**

**global posX**

**global posY**

**myPen.up()**

**myPen.left(180)**

**myPen.forward(posX)**

**myPen.left(90)**

**myPen.forward(18)**

**myPen.left(45)**

**#myPen.left(90)**

**myPen.down()**

**posX = 0**

**posY = posX + 18**

**for row in range (5) :**

**#for row in range (9)**

**for column in range(9) :**

**drawPixel(pixelMemory[pixelAddress])**

**pixelAddress += 1**

**newRow()**

**SAMPLE PROGRAM #1 - Syntax Error**

import turtle

myPen = turtle.Turtle()

circleColors = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down(

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 0

for circleIndex in range(3) :

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

**SAMPLE PROGRAM #2 - Run-time Error**

import turtle

myPen = turtle.Turtle()

circleColours = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down()

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 1

for circleIndex in range(4) :

print(circleNumber)

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

**SAMPLE PROGRAM #3 - Logic Error**

import turtle

myPen = turtle.Turtle()

circleColours = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down()

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

numOfCircles = 2

for circleIndex in range(2) :

circleNumber = numOfCircles - circleIndex

print(circleNumber)

drawCircle(circleColours[circleNumber])