Mohawk_Pygame: Sample Exercises

These are exercises from my handouts for Sheridan's *Programming Principles* course, adapted to use the mohawk_pygame.py module. Content covered includes basic I/O, variables, types, expressions, if, while and functions. I have not yet managed to work in mouse and keyboard control so there are no exercises on those topics here. There are solutions to some of these exercises in the accompanying folder.

Basic Drawing

1. Try out all the mohawk_pygame shape and text drawing commands and make sure you understand how they work. Try using the extra width parameter for the shapes. Try to make mistakes or create errors. What happens if you use negative or fractional values for some of the numbers? What happens if you leave out a parameter?

2. Use the mohawk_pygame shape and text drawing commands to create a logo for the course, the BAScM program or the college. Your logo should blend text with multiple shapes and colors. Try to use `mpg.pause` to add some creative timing effects as well. Hand in whatever you create to the First Pygame Program drop box (this exercise will not be graded).

Variables and Expressions

3. **Using Variables**
   Write a program that starts by having a dialog with the user. Ask them for a point (x,y) a color name and a radius. Then open the mohawk_pygame screen and draw the circle they specified. Don't exit until they have pressed enter in the shell.

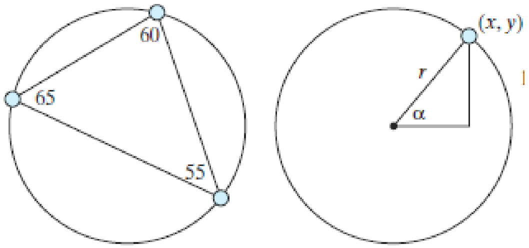
4. **Using Expressions**
   Modify exercise 3 so that instead of entering a radius, they enter an area for the circle (a `float`). Then compute the radius from the area using by rearranging the formula $A = \pi r^2$.

Using the Math Module

1. (Challenging) Write a program that generates three random points on a circle centered at (0, 0) with radius 100 and draw the resulting triangle and circle in mohawk_pygame.

   Python contains a module for generating pseudo-random numbers. Use `import random`, then `random.randint(a, b)` to generate a random integer from a to b (inclusive).

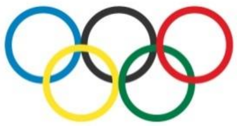
   Hint: Generate a random angle α in radians between 0 and 2π, as shown in the figure below and the point determined by this angle is $(r\cos(\alpha), r\sin(\alpha))$.



   Hint: When drawing, add a constant value (like 200) to every point (circle center point and the three points you generate) to move the circle and triangle onto a part of the screen you can see.

Functions

2. Write a function (using the mohawk_pygame template) that takes a pair of x and y coordinates and then draws the olympic logo on the drawing surface using the coordinates given as the middle of the black circle.



3. Modify the function from part 4 so that it also takes a radius parameter and draws the Olympic logo using that radius.

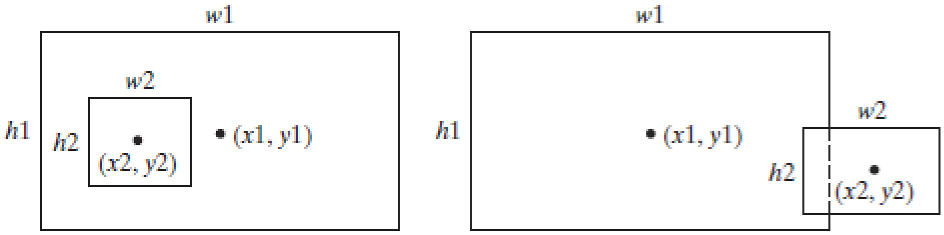
4. Write a function called that calls the function from step 4 (or 5) three times to place the Olympic logo in random places (and random sizes) on the screen.

5. (Random Numbers and mohawk_pygame) Write a function to display a random shape in the graphics console. Pick the size, location, and colors randomly (use `mpg.set_color(r, g, b)` where `r`, `g` and `b` are replaced with integers in the range 0 to 255.). Choose randomly whether to draw an ellipse or a rectangle. Use a `for` statement to call the function 100 times with a delay of 0.1 seconds between each call.

If Statements

6. (Arithmetic and If) Write a program that prompts the user to enter a point (x, y) and checks whether the point is within the circle centered at (0, 0) with radius 10. For example, (4, 5) is inside the circle and (9, 9) is outside the circle. Use mohawk_pygame to draw the circle and the point so you can check whether or not your program got it right.

7. Write a function that takes as parameters the center *x*-, *y*-coordinates, width, and height of two rectangles and determines whether the second rectangle is inside the first or overlaps with the first, as shown the figure below (the left rectangles are inside, the right rectangles overlap). Test your function to cover all cases and record the tests you performed. Write code to allow a user to enter all the values and then call the function and pop up a mohawk_pygame window to draw the rectangles the user entered and visually check your results.



Loops

8. Write a mohawk_pygame program that moves a shape from the far left to the far right of the mohawk_pygame window using a counted loop. Can you make the shape grow or shrink as it moves? Can you make it fade out as it moves? Can you make another shape move from right to left at the same time?

9. Using mohawk_pygame, create a function using nested loops that displays a checkerboard. The function should accept the number of rows and columns as parameters.