Announcements

Reminders:
Program 7 – due tomorrow by 11:55pm

Using Graphics

- We learned about Turtle
  - Many other graphics libraries available
- Introducing simplegraphics
  - Not a built-in library
  - You need to have 2 files in the same folder as your program
    - simplegraphics.py
    - cs1graphics.py

My First Graphics Program

```python
from simplegraphics import *

def main():
    open_canvas(500, 500)
    draw_circle(250, 250, 100)
    close_canvas_after_click()

main()
```
Another Graphics Program

```python
from simplegraphics import *

def draw_bullseye(xcenter, ycenter, color):
    set_color(color)
    draw_filled_circle(xcenter, ycenter, 50)
    set_color('black')
    draw_filled_circle(xcenter, ycenter, 40)
    set_color('black')
    draw_filled_circle(xcenter, ycenter, 30)
    set_color('black')
    draw_filled_circle(xcenter, ycenter, 20)

def main():
    # create a canvas and draw 2 overlapping bullseyes on it
    open_canvas(300, 300)
    draw_bullseye(100, 100, 'blue')
    draw_bullseye(100, 100, 'orange')
    close_canvas_after_click()

main()
```

Adding On

```python
from simplegraphics import *

def draw_bullseye(xcenter, ycenter, color, outline):
    set_color(color)
    draw_filled_circle(xcenter, ycenter, 50)
    set_color('black')
    draw_filled_circle(xcenter, ycenter, 40)
    set_color('black')
    draw_filled_circle(xcenter, ycenter, 30)
    set_color('black')
    draw_filled_circle(xcenter, ycenter, 20)

def main():
    # create a canvas and draw 2 overlapping bullseyes on it
    open_canvas(300, 300)
    draw_bullseye(100, 100, 'blue', 'black')
    draw_bullseye(100, 100, 'black', 'orange')
    wait_for_click()
    close_canvas()

main()
```

Refer to the simplegraphics handout for all available functions for this library.

Practice

1. Modify Nov17.py (Public directory) to first take in input from the user for the x, y coordinates of the bullseye's center.
2. Next, change the code to now choose the center of the bullseye using a mouse click. (You'll need to refer to the sheet I just gave you about the functions.)
3. Last, modify the code to now choose the center and the radius of the bullseye with 2 mouse clicks (the first gives you the center, second click chooses a point on the border of the bullseye from which you can compute the radius.
   - You'll need to use the distance formula for this one
     - \[ \text{distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \]
     - Use the math module (import math)
     - math.sqrt(x) where x is a number (integer or floating-point)

Adding Graphics to Tic-Tac-Toe

More fun to be able to just click on the location of where you want to make your move, rather than type in row and column.

What modifications to our existing Tic-Tac-Toe game do we need to make to be able to allow users to just click on the board to make their move?
Extending Our Board

• If we wanted to create an 6 x 6 this time, what would we need to do differently?
• What functions do we need still?

Practice

• Create an 8 x 8 board (using graphics).
• Write code to correctly determine in which row and column a user clicks the mouse.
• Once a board location has been chosen, mark it somehow so that the user won’t be allowed to click in it again.
• You may mark previously selected locations with any shape you’d like.
• Write a function to determine if there are still moves left, and if so, you should continue to let the user click on the board.