

15-112 Fall 2022 Lecture 3

Quiz 1A

35 minutes

Name: \_\_\_\_\_

Andrew ID: \_\_\_\_\_@andrew.cmu.edu

Section: \_\_\_\_\_

- You may not use any books, notes, or electronic devices during this quiz.
- You may not ask questions about the quiz except for language clarifications.
- Show your work on the quiz (not scratch paper) to receive credit.
- If you use scratch paper, you must submit it with your andrew id on it, and we will ignore it.
- All code samples run without crashing unless we state otherwise. Assume any imports are already included as required.
- Do not use these topics: lists, sets/dictionaries, recursion.
- You may use `almostEqual()` and `rounded()` without writing them. You must write everything else.

Do not write below here

Question	Points	Score
1. FR: Blue-or-Red Ring	50	
2. FR: Gold Square	50	
3. Bonus	5 (bonus)	
TOTAL	100	

**1. Free Response: Blue-or-Red Ring [50 pts]**

Write an app such that:

1. At first, it displays a blue "ring" in the center of the canvas. To draw the ring, draw a circle of radius 40 with a fill of None and a blue border with `borderWidth` 20.
2. If the user presses the mouse inside the colored part of the ring (the blue part, at first), then the colored part toggles from blue to red or from red to blue.
3. If the user presses in the inner empty part of the ring, then the ring becomes orange and follows the mouse as it is dragged, centered on the mouse position. When the mouse is released, the ring returns to the color it was before it was dragged.
  - You only have to move the ring when the mouse is dragged, not when it is pressed or released.
4. Mouse presses entirely outside the ring are ignored.

Hint: to solve this, you will need to write `onAppStart`, `redrawAll`, `onMousePress`, `onMouseDrag`, and `onMouseRelease`. You may also want to write a helper function to find the distance between two points.

This page is blank (for your Red-or-Blue Ring solution, if needed).

**2. Free Response: Gold Square [50 pts]**

Write an app such that:

1. At first, a 20x20 gold square is drawn so it just fits inside the top-left corner of the canvas.
2. Every 1/2 second, the gold square moves to the next corner, at first moving clockwise (so after the top-left corner, it moves to the top-right, then bottom-right, then bottom-left).
3. Each time the user presses 'r', the gold square reverses its direction, toggling from clockwise to counterclockwise or vice versa.
4. When the user presses and holds the 'p' key, the gold square pauses and does not move. When the user releases the 'p' key, the gold square resumes moving in the same direction it was moving before the pause.
5. If the user presses the 's' key while paused, the gold square moves one step in the direction it was moving before the pause. However, the 'r' key is ignored while paused.

Hint: to solve this, you will need to write `onAppStart`, `redrawAll`, `onKeyPress`, `onKeyRelease`, and `onStep`. You may also want to use `app.stepsPerSecond`, and to write a helper function called something like `takeStep`. Finally, while not required, it may help to number the corners in some way from 0-3, then write a function that takes a number from 0-3 and returns the x,y coordinates of the top-left corner of the gold square if it is in that particular corner.

This page is blank (for your Gold Square solutions, if needed).

**3. Bonus [5 pts]**

Indicate what these print. Place your answers (and nothing else) in the box next to each block of code.

```
def z(x):
    y,z = 0,x
    while x > 0: y,x = y+z, x-1
    return y

def bonus_ct1(s):
    t, s = 0, s.upper()
    for c in s: t += z(ord(c) - ord(min(s)))
    return t

print(bonus_ct1('DbGc'))
```



```
def bonus_ct2(s):
    t,e = '',1
    for c in s:
        d = (int(c) if c.isdigit() else ord(c) - ord('e') if
             c.isalpha() else ord(c) - ord(s[-1]))
        if not bool(d):
            t += chr(ord(c) + (e if c != s[-1] else 0))
            e += 1
    return t

print(bonus_ct2('I saw 10 bees about 20 days ago!'))
```

