

15-112 Spring 2023 Quiz 7

Up to 20 + 10 minutes (finish within 20 minutes for 20% proficiency bonus). No calculators, no notes, no books, no computers. Show your work!

Do not use try/except, dictionaries, or sets on this quiz

1. (3 points) **Short Answers:** Fill in each blank with Model, View, or Controller:

- (a) The _____ draws the app using the values in the model.
- (b) The _____ contains all the data we need for the animation.
- (c) The _____ responds to the keyboard, mouse, timer, and other events and updates the model.

2. (5 points) **MVC: Indicate the violations:**

The following code is a failed attempt to implement an animation. Ignoring the possible run-time/syntax errors, can you identify MVC violations? Write down the **line numbers** where MVC violations occur.

```
1  def appStarted(app):
2      app.timerDelay = 100
3      app.paused = False
4      app.dots = [10]
5
6  def keyPressed(app, event):
7      if event.key == "p":
8          if app.paused:
9              app.paused = False
10         else:
11             app.paused = True
12             canvas.create_text(app.width//2, app.height//2,
13                                 text="Game paused")
14
15  def redrawAll(app, canvas):
16      if app.paused == False:
17          r = app.dots[-1]
18          app.cx = random.randint(0,app.width)
19          app.cy = random.randint(0,app.height)
20          canvas.create_oval(app.cx - r, app.cy - r,
21                              app.cx + r, app.cy + r, fill="red")
22      else:
23          app.r = random.randint(10, 20) # get a random integer between 10 and 20
24          canvas.create_oval(app.cx - r, app.cy -r,
25                              app.cx + r, app.cy + r, fill="blue")
26          app.dots.append(r)
27
28  runApp(width=800, height=800)
```

3. (12 points) **Free response: Animation** Assuming the `runApp()` function is already written for you, write `appStarted`, `keyPressed`, `mousePressed`, `redrawAll`, and `timerFired` so that when the animation is first run:

1. One circle is randomly positioned on the screen. Initially, the circle has a radius of 40 pixels and random color (chosen from `['red', 'green', 'blue']`).
2. An integer number is displayed on the top-left corner that counts down in seconds starting from 20. (1 point)
3. An integer number is displayed in the bottom-right corner that counts the points obtained so far. (1 point)

Game play proceeds as such:

1. Every two seconds, the color is randomized. This means a new random color for the circle is chosen using the same three colors (red, green, and blue). (2 points)
2. If the user clicks inside the circle when its color is blue, one point is obtained. (2 points)
3. If the user clicks **outside the circle** or inside the circle **when its color is not blue**, the game becomes harder: the current radius of the circle halves (first, it is 40 pixels, then 20 pixels, then 10 pixels, and finally 5 pixels). (2 points)
4. The user wins when 5 points are obtained. After winning, the text "You Win!" is displayed in the center of the screen. (2 points)
5. The game ends with a loss when the user makes four mistakes (clicks outside the circle or inside a non-blue circle) OR the countdown reaches 0. When the game is over with a loss, the screen is cleared (the circle is removed), and the text "Game over!" is displayed in the center of the screen. (2 points)

Make reasonable assumptions for anything not specified here. Do not hardcode values for `app.width` or `app.height`. We recommend that to save time writing, you abbreviate `canvas`, `event`, and `app`: use `c`, `e` and `a`, respectively.

Below is some starter code for the `appStarted` function. You can expand it and write the other functions as needed.

```
from cmu_112_graphics import *
import random

def appStarted(a):
    # Configuration
    a.radius = 40 # initial radius
    # Initialize random color
    a.randomColor = random.choice(["red", "yellow", "blue"]) # pick one at random
    # Initialize circle coordinates
    cx = random.randint(0, a.width) # random initial x
    cy = random.randint(0, a.height) # random initial y
    a.circle = (cx, cy)
    # You should probably initialize other things in appStarted
    # You also must write the other functions: timerFired, mousePressed, keyPressed, redrawAll
```

Answer space for Question 3