

fullName: _____ andrewID: _____ recitationLetter: _____

15-112 S23

Quiz4 version D (30 min)

You **MUST** stop writing and hand in this **entire** quiz when instructed in lecture.

- You may not unstaple any pages.
- Failure to hand in an intact quiz will be considered cheating. Discussing the quiz with anyone in any way, even briefly, is cheating. (You may discuss it only once the quiz has been posted to the course website.)
- You may not use your own scrap paper. If you must use additional scrap paper, raise your hand and we will provide some. You must hand any scrap paper in with your paper quiz, and we will not grade it.
- Please try to limit questions so as to not distract your peers. **We will answer two questions at most per person.** If you are unsure how to interpret a problem, take your best guess.
- Unless otherwise stated, you may not use any concepts (including builtin functions) which we have not covered in the notes in weeks 1-4. You may not use tuples, dictionaries, sets, or recursion.
- Assume `almostEqual(x, y)` and `rounded(n)` are both supplied for you. You must write all other helper functions you wish to use, unless we specify otherwise.

Multiple Choice 1 [8pts]

Which of the following would be considered an MVC error?

Select ALL that apply. (fill in at least one square).

- 1. Calling drawRect in onMousePress
- 2. Setting numberOfSquares (an integer) to 10 in onMousePress
- 3. Setting the variable app.x (an integer) to 5 in onMousePress
- 4. Setting the variable petName (a string) to "Chee" in redrawAll
- 5. Incrementing the variable app.x (an integer) in redrawAll
- 6. Calling drawCircle in redrawAll
- 7. Using a for loop in redrawAll
- 8. Using a for loop in onStep

Multiple Choice 2 [4pts]

Which of the following images is drawn by this code?
Assume a 400x400 canvas. Select ONE.

```
from cmu_graphics import *
```

```
def redrawAll(app):
```

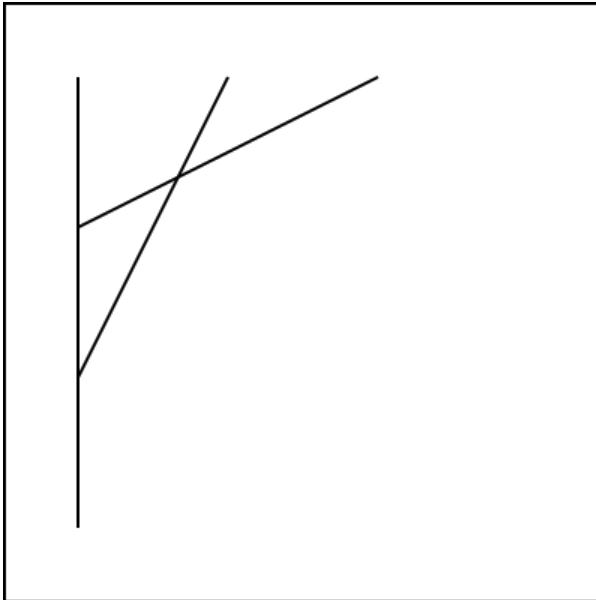
```
    space = 100
```

```
    for i in range(3):
```

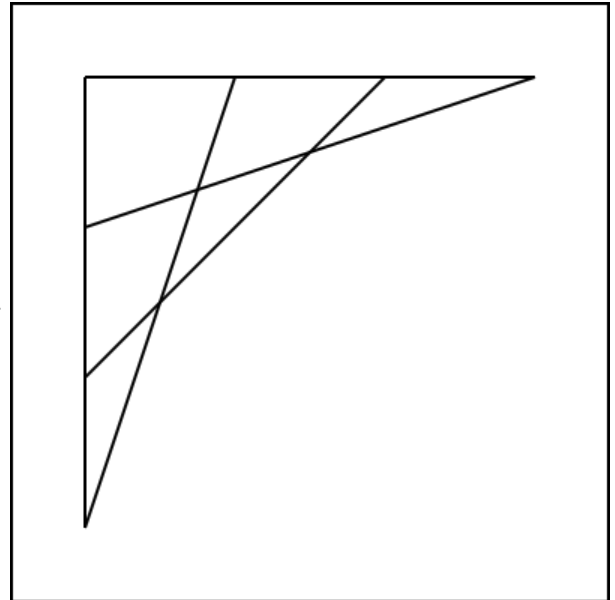
```
        drawLine(50, 350 - (i * space), 50 + (i * space), 50)
```

```
runApp()
```

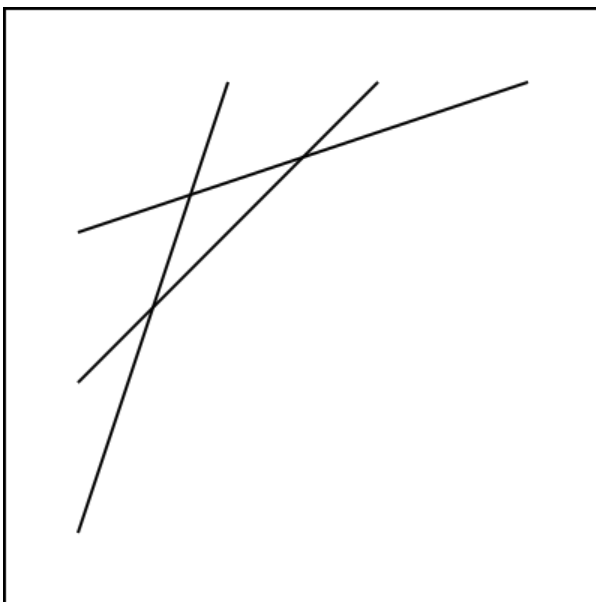
A.



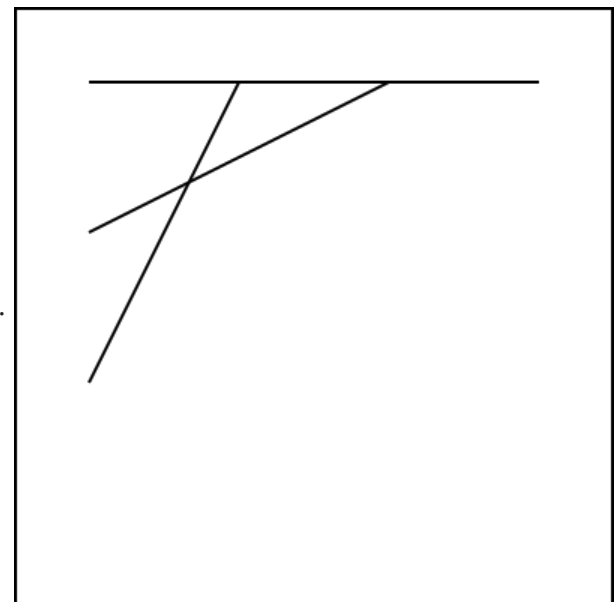
B.



C.



D.



Free Response 1: Square animation [44pts]

Write an animation with the following features.

- When the app starts, a square with side lengths of 50 pixels is drawn centered on the canvas.
- When the mouse is moved, the square's location changes so that it is centered on the mouse cursor. Note: Mouse drag events (i.e. when the mouse button is held down) do not move the square.
- When the mouse is clicked, the square's side length increases by 10 pixels
- At every `onStep(app)` call, if the square's side length is larger than 50, the side length decreases by 1.

Notes:

- Do not hardcode for a 400x400 canvas. However, you may assume the canvas is square, and at least 100x100 pixels.
- You will be penalized if your code results in an MVC violation.
- Make reasonable choices for anything not specified above (for example, the color of the square).
- Please write your code using the provided function headers for `onAppStart`, `redrawAll`, `onMousePress`, `onMouseMove`, and `onStep`. As usual, you may write and use additional helper functions if you wish. (You may also add other event functions if you feel you must, but this would probably be a bad idea.)

Begin your FR1 answer here:

```
from cmu_graphics import *  
  
def onAppStart(app):
```

Continue your FR1 answer here

```
def redrawAll(app):
```

```
def onMouseMove(app, mouseX, mouseY):
```

Continue your FR1 answer here

```
def onMousePress(app, mouseX, mouseY):
```

```
def onStep(app):
```

```
runApp()
```

Free Response 2: Dot Walk Animation [44pts]

Write an animation with the following features.

- When the app starts:
 - a green dot with radius 42 is drawn with its left edge touching the left edge of the canvas, and vertically centered at $y = 200$
 - The label "GO" is drawn near the top of the canvas, horizontally centered
- When a key is pressed:
 - The label changes from "GO" to that key. For example, if an "m" is pressed, the label reads "m"
 - If the user presses "x" AND the dot is green, the dot immediately moves to the right by 10 pixels and becomes red.
 - If the user presses "y" AND the dot is red, the dot immediately moves to the right by 10 pixels and becomes green.
 - If the user presses "r" the animation resets (so the dot is green and touching the left edge, and the label says "GO")
 - Except in the cases above, no key presses or mouse actions will move the dot (though each keypress updates the label)
- When the dot's right edge touches or moves beyond the right edge of the canvas, the label immediately changes to "DONE" (and does not change again until the app is reset with 'r'). However, "x" and "y" presses will continue to move the dot beyond the edge of the canvas.

Notes:

- Do not hardcode for a 400x400 canvas. However, you may assume the canvas is square, and at least 100x100 pixels.
- You will be penalized if your code results in an MVC violation
- Make reasonable choices for anything not specified above (for example, font size).
- Please write your code using the provided function headers for `onAppStart`, `redrawAll`, and `onKeyPress`. As usual, you may write and use additional helper functions if you wish. (You may also add other event functions if you feel you must, but this would probably be a bad idea.)

Begin your FR2 answer on the next page:

Begin your FR2 answer here

```
from cmu_graphics import *
```

```
def onAppStart(app):
```

```
def redrawAll(app):
```


Continue your FR2 answer here

```
def onKeyPress(app, key):
```

```
runApp()
```

bonusCT [1pt]

Sketch what this code draws in the box provided. Assume a 400x400 canvas.

```
from cmu_graphics import *

def redrawAll(app):
    for i in range(7):
        drawLine(max(50, (i*100)-250),
                 min(50 + (i * 100),350),
                 min(50 + (i+1) * 100,350),
                 50)
        drawCircle(max(50, (i*100)-250),
                  min(50 + (i * 100),350),5*(i+1),
                  fill = None, border = 'black')

runApp()
```

