

15-112 Spring 2023 Lecture 3/4

Quiz 4

5 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: sets/dictionaries, recursion.
- You may use `almostEqual()` and `rounded()` without writing them. You must write everything else.

1. FitB (Fill in the Blank) [100 pts]

There is only one part, FitB (Fill in the Blank), in this quiz.

Place your answer in the box.

Enter one line for each blank. So a problem with two blanks should have two lines of text in your answer.

FitB 1:

Fill in the blank so that this only calls `doSomething(app)` when the user is holding down the right arrow key but not holding down the left arrow key:

```
def onKeyHold(app, keys):  
    if _____:  
        doSomething(app)
```

FitB 2:

Fill in these two blanks with the missing function call so that this function works correctly:

```
def getRadiusAndAngleToEndpoint(cx, cy, targetX, targetY):  
    radius = distance(cx, cy, targetX, targetY)  
    angle = math.degrees(math.atan2(_____, _____)) % 360  
    return (radius, angle)
```

This problem has two blanks, so you should enter two lines of text in your answer (one for each blank).

FitB 3:

Fill in the two blanks so that this function works correctly (Hint -- recall that theta is in degrees):

```
def getRadiusEndpoint(cx, cy, r, theta):  
    return ( _____ ,  
            _____ )
```

This problem has two blanks, so you should enter two lines of text in your answer (one for each blank).

FitB 4:

Fill in the blank so that this code draws a polygon using the values in the variable 'points' (do not hardcode the values):

```
def redrawAll(app):  
    points = [100, 100, 50, 200, 300, 300, 250, 50]  
    drawPolygon( _____ , fill='cyan', border='black')
```

2. Bonus [3 pts]

Indicate what this prints. Place your answer (and nothing else) in the box.

```
def bonusCt(s):  
    def f(n): return (2*n - 1 + f(n-1)) if bool(n) else 0  
    return sum([f(f(ord(c)-ord(s[0]))) for c in s])  
  
print(bonusCt('abcba'))
```