15-112 Spring 2019 Quiz 8

Up to 20 minutes. No calculators, no notes, no books, no computers. Show your work!

Do not unstaple this quiz

You may not use recursion in your solutions

1. (30 points) For each question, fill in the circle for **all** of the answers that are correct. **Some questions may have more than one correct answer.**

(a) What is a Class in Python?
- ○ A method
- ○ A template
- ○ A specific item
- ○ A specialized function

(b) What is an Instance in Python?
- ○ A method
- ○ A template
- ○ A specific item
- ○ The opposite of a class

(c) What is a Constructor in Python?
- ○ A method that makes an instance
- ○ The place where attributes are first set up
- ○ A generic object that makes an instance
- ○ A method called to create new classes

(d) Which of the following could be a superclass of Fruit?
- ○ Apple
- ○ StringInstrument
- ○ Vegetable
- ○ Food

(e) Which of the following statements about class attributes are **True**?
- ○ They can be accessed from the class itself
- ○ They cannot be modified
- ○ They are defined in `__init__`
- ○ They can be accessed from any instance of the class

(f) In `def __eq__(self, other):`, "other" is...
- ○ Never a copy of self
- ○ Compared with self when using ==
- ○ Always the same type as self
- ○ A method for changing the class of an object

(g) Which of the following statements about OOPy animation are **True**?
- ○ It helps us organize animations
- ○ Objects can have MVC components
- ○ Objects can create other objects
- ○ It is an MVC violation to create objects in timerFired

(h) If we have a class called ZooAnimal and we want to store them in a set, which of the following properties should we **avoid** using in our hash method?
- ○ The animal's birthday
- ○ Whether the animal is sleeping
- ○ The animal's age
- ○ The animal's species

2. (40 points) **Free Response:** Write the classes Person and Student so that they pass the following test cases. For full credit you must use inheritance appropriately. You should probably write your answer on the following page so you have plenty of space!

Hint: Use 's' instead of 'self' to save time when writing!

```python
# A Person has one property: whether or not they are currently sleeping.
p1 = Person(False)
assert(str(p1) == "Person(False)")
# A person can sleep and wake up
assert(p1.sleep() == "Sleeping!")
assert(str(p1) == "Person(True)")

assert(p1.wake() == "Awake!")
assert(str(p1) == "Person(False)")

assert(str(Person(True)) == "Person(True)")

# A Student is a Person that needs to complete an assignment.
# A Student cannot work if they are asleep
# 1st param is sleeping state; the 2nd is whether work complete.
s1 = Student(True, False)

# list the sleeping state first, then whether the assignment is complete
assert(str(s1) == "Student(True,False)") # asleep, work incomplete

# Can't work if sleeping!
assert(s1.work() == "Can't work!")
assert(str(s1) == "Student(True,False)") # still asleep, work incomplete

# Able to work after waking up
assert(s1.wake() == "Awake!")
assert(str(s1) == "Student(False,False)") # s1 is awake, work incomplete
assert(s1.work() == "Done!")
assert(str(s1) == "Student(False,True)") # s1 is awake, work complete
assert(s1.sleep() == "Sleeping!")
assert(str(s1) == "Student(True,True)") # s1 asleep, work complete

# Check for inheritance
assert(isinstance(s1, Person) == True)
assert(isinstance(p1, Student) == False)
```

YOU MAY CONTINUE WRITING CODE ON THIS PAGE.

THERE IS ALSO A PROBLEM ON THE NEXT PAGE! DON'T MISS IT!

3. (30 points) **Free Response:** The following code is part of an animation that draws a Dot with a radius of 25 pixels at a random location every two seconds. When a dot is clicked, it is deleted by removing it from the list. Write the Dot class (and nothing else!) so that the animation is complete. Read the code carefully so that you know how the Dot class should behave! Note that we use e, c, and d to replace event, canvas, and data, respectively.

Note: We have provided the helper function `distance(x1, y1, x2, y2)` if you wish to use it.

Hint: The easiest solution only requires three methods.

```python
import random

def distance(x1, y1, x2, y2):
    return ((x1-x2)**2 + (y1-y2)**2)**0.5

def init(d):
    d.dots = []
    d.timerCount = 0

def mousePressed(e, d):
    i = 0
    while i < len(d.dots):
        dot = d.dots[i]
        if dot.clicked(e.x, e.y):
            d.dots.pop(i)
        else:
            i += 1

def timerFired(d):
    d.timerCount += 1
    if d.timerCount % 20 == 0:
        x = random.randint(0, d.width)
        y = random.randint(0, d.height)
        d.dots.append(Dot(x, y))

def redrawAll(c, d):
    for dot in d.dots:
        dot.draw(c)
```

YOU MAY CONTINUE WRITING CODE ON THIS PAGE.

YOU MAY CONTINUE WRITING CODE ON THIS PAGE.