Name: _____ Recitation: _____ Andrew Id: _____

15-112 Spring 2019 Practice Quiz 8

Up to 25 minutes. No calculators, no notes, no books, no computers. Show your work!

Do not use recursion on this quiz.

1. (5 points) **Short Answer:** Name one benefit of using inheritance

2. (5 points) **Short Answer:** There are two methods we use to define the string representation of an object.

   `__str__` and `__repr__`

   Which do we usually use and why?

3. (40 points) **Free Response: Birds**

   Write the necessary classes and methods so that the following test function passes. You may not hardcode any test cases. You must use good Object Oriented Design and inheritance principles to receive full credit.

```
def testOOP():
    zazu = Bird("eep", True)
    assert(zazu.squawk() == "eep")
    assert(zazu.canFly)
    assert(str(zazu) == "just a Bird lol")

    # hint: what kind of attribute is this?
    assert(not Bird.isMigrating)

    kowalski = Penguin("kung fu")
    assert(isinstance(kowalski, Bird))
    assert(kowalski.squawk() == "bloop")
    assert(not kowalski.canFly)
    assert (kowalski == Penguin("kung fu"))
    assert(str(kowalski) == "Penguin who knows kung fu")

    skipper = Penguin("taekwondo")
    assert(skipper != kowalski)
    assert(skipper != "dont crash here :^)")
    assert(str(skipper) == "Penguin who knows taekwondo")

    # hint: what kind of method is this?
    Bird.winterIsComing()
    assert(Bird.isMigrating)
```

4. (50 points) **Free Response: OOPy Space**

Space - the final frontier. In it, we have discovered the salubrious solar system - a collection of planets known for their health-giving properties - and our scientists insist that we model it for research purposes. Wary of the work ahead but undeterred, we begin what may be our best work yet.

In modeling the salubrious solar system our scientists have informed us of certain properties we should satisfy:

- It has a single sun with mass ranging between 1000 and 1500 (random) and a radius varying between 25 and 50 (also random). The sun is in the middle of the screen, and does not move.
- Initially, 6 planets float in space, generated in the following way:
  - Planets are generated in the top left corner of the canvas
  - They have a random x and y velocity (in any direction, with speed between 5 and 10)
  - And a random mass between 200 and 500
  - And a random radius between 5 and 20.
- Every timer fired, the planets move in the direction defined by their x and y velocities.
- If a planet collides with the sun, the sun gets hungry and eats it (realistic, I know) and gains half of the planet's radius and mass and a new planet is spawned.
- Every second, planets and the sun change color from white to some random color (either pink, purple, or green), or from non-white back to white. Also, display the planet/sun's mass in the middle of it.

In order to receive full credit for this problem, you must have a parent class from which both the Sun and Planet classes inherit from, and use OOP in a meaningful way in your animation. You may assume that the run function has been written for you.