Quiz 5 3/17/14 Name:_____________________________ ___/20

1. Fill in some code for a Cat class, following guidelines given through comments.

```java
public class Cat {
   // instance variables
   String name;
   int age;
   int lives; // number of lives remaining

   // constructor: Creates a y-year-old Cat named x with z lives.
   public Cat (String x, int y, int z)
   {
      name = x;
      age = y;
      lives = z;
   }

   // Another constructor: Creates a Cat named x.
   // This Cat has default age (0), default lives (9).
   public Cat (String x)
   {
      // WRITE METHOD DEFINITION HERE: ******************************
   }

   //******** Assume the following methods are also implemented ********
   // death(): decreases the number of lives by 1.
   // getAge(): returns the Cat’s age
   // getLives(): returns the Cat’s remaining lives
   // toString(): returns a String representing this Cat
   //************************************************************************
}
```

2) Write client code that uses the Cat class:

a) Instantiate an Cat object with name “Luca”, 10 years old with 6 lives. Assign it to a variable named profsCat

b) Instantiate another object of the Cat class with name “Tuna”, using the 2nd constructor (so it will be 0 years old with 9 lives – a kitten!). Assign it to a variable named otherCat

c) Print the info of profsCat and otherCat (use the toString() method).

d) Print the average age of profsCat and otherCat (use the getAge() method). Note that your code should work for no matter how profsCat and otherCat are initialized.; thus System.out.println(5) is NOT the answer, even though it would give the correct value for our example.
1. Fill in some code for a Cat class, following guidelines given through comments.

```java
public class Cat {
    // instance variables
    String name;
    int age;
    int lives; // number of lives remaining

    // constructor: Creates a y-year-old Cat named x with z lives.
    public Cat (String x, int y, int z) {
        name = x;
        age = y;
        lives = z;
    }

    // toString(): returns a String representing this Cat
    public String toString() {
        // WRITE METHOD DEFINITION HERE: *****************************************
    }
}

//******** Assume the following methods are also implemented *********
// death(): decreases the number of lives by 1.
// getAge(): returns the Cat’s age
// getLives(): returns the Cat’s remaining lives
//*************************************************************************

2) Write client code that uses the Cat class:
   a) Instantiate a Cat object with name “Tony”, 5 years old with 8 lives. Assign it to a
      variable named cat1

      b) Instantiate another Cat object with name, age, and number of lives of your choice.
         Assign it to a variable named cat2

      c) Print the info of cat1 and cat2 (use the toString() method).

      d) Print the average number of lives of cat1 and cat2 (use the getLives() method).
         Note that your code should work for no matter how cat1 and cat2 are initialized; thus something like
         System.out.println(5) is NOT the answer, even if it gives the correct value for your example.