CS211 Fall 2012

Dr. Kinga Dobolyi

Exam 1

Student Name: ____

Student G#: ____

Student signature for Honor Code:

____

Part 1: Short Answer.

1. Draw what memory looks like after each of the following statement blocks (sub-parts are independent). Show garbage collection by crossing out items (i.e. do no erase anything for this question): (17 points)

   a.
```
String s = new String("Kinga");
String d = new String("David");
int x = 4;
s = "John";
//show garbage collection if it happened on this line,
//assuming there are lines after
```

   b.
```
Person p;
p = new Person("Sally", 19);
//assume Person stores a String name and int age
//show garbage collection if it happened on this line,
//assuming there are lines after
```

   c.
```
Integer i = new Integer(44);
float x = 3;
```

2. What are the results of the following ***expressions***? Give the Java type and the value of the expression. If there are multiple expressions/statements in the question, give the value of the last expression. Assume all code compiles.
(14 pts)

a. **`1 == 2`** __________________ __________________

b. **`3 / ((float)2)`** __________________ __________________

c. **`x = false;`**

`“3” + x` __________________ __________________

d. **`x = 4;`**
`x++;`

`x / 4 + 3` __________________ __________________

e. **`true && p1==p1`** __________________ __________________

f. **`String s1 =“ki”;`**

`s1.charAt[1]` __________________ __________________

g. **`“1” + 3`** __________________ __________________

3. Give the output of the following code (22 points):

```
public class Driver{
     private static Person person = new Person("n", 0);

     //Assume each Person stores a String name
     //and an int age, as public fields
     //When printing out Person-s, print their
     //name and age, i.e., “Kinga 23”
     public static void main(String[] args){
          int x = 2;
          String name = "John";
          Person p1 = new Person(name, 11);
          Person p2 = p1;
          Person p3 = new Person(name, 11);

          System.out.println(x);
          System.out.println(name.toString());
          System.out.println(p1);
          System.out.println(p2);
          System.out.println(p3);
          System.out.println(person);
          System.out.println(p1 == p3);
          System.out.println(p1 == p2);

          p3 = change(name, x, p1);

          System.out.println(x);
          System.out.println(name.toString());
          System.out.println(p1);
          System.out.println(p2);
          System.out.println(p3);
          System.out.println(person);
     }
     public static Person change(String name, int y,
               Person p){
          person.name = name;
          name = "Bob";
          p.name = name;
          y++;
          p.age = y;
          p = new Person("Liz", 10);
          return p;
     }
}
```

(answer here)

4. A. Write a public class called **`Animal`** according to the following specifications:

    a. The class should have two mutable attributes, a **`String`** *species* and a primitive integer *weight*.

    b. The class should have a default constructor which initializes both fields.

    c. The class should have another constructor that initializes both fields with arguments passed in through the constructor

    d. The class should have a **`toString`** method that returns a nicely formatted (your choosing) representation of the contents of an **`Animal`** object as a **`String`** object.

    e. The class should have a method to change the *species* of the Animal to an incoming argument

    f. The class should always ensure that the *weight* is a nonzero integer, and that the length of the *species* field is at least 1, but less than 20 (**`String`** has a **`size()`** method that returns the length of the **`String`**).

    g. The class should use good object-oriented design (22 pts)

(use this space for 6 if you need it)

Part 2: True/False (circle one), and for full credit ***justify*** all of your choices. For this question, assume there is a **`Person`** class, and a **`Driver`** class that has a **`main`** method, that creates **`Person`** objects **`p1`** and **`p2`**. These two classes have been compiled and are in the same directory. (8 pts)

1. A public static method of **`Person`** can be called on **`p1`**

   1.TRUE FALSE

2. A public static method of **`Person`** can be called on **`Person`** (the class) in **`main`**

   2.TRUE FALSE

3. A static method can update both static and non-static attributes of the same class

   3.TRUE FALSE

4. A public static method can call other public static methods within its body, but not private static ones

   4.TRUE FALSE

5. A private method can only access private attributes within the same class

   5.TRUE FALSE

6. The **`main`** method can call a private method of **`Person`** on **`p2`**

   6.TRUE FALSE

7. The **`equals`** method can be called on **`p1`** with **`p2`** as an argument

   7.TRUE FALSE

8. I must import **`java.util.Scanner`** before using a **`Scanner`** object

   8.TRUE FALSE