Name: _____

CS 141: Introduction to (Java) Programming: Exam 3

Orr • Willamette University • Fall 2011

Page 1:	(max 24)	Page 4-5:	(max 26)
Page 2-3:	(max 32)	Page 6:	(max 18)
Total:		(max 100)	

1. (4 pts) What does an object reference specify?
   a. The size of an object.
   b. The instance variables of an object.
   c. The default value of an object.
   d. The location of an object.
2. (4 pts) An object stores its data in:
   a. Methods.
   b. Classes.
   c. Instance variables.
   d. It doesn't store any data.
3. (4 pts) Under what conditions can you overload method names?
   a. The parameter types or the number of parameters must be different.
   b. The parameter types must be identical.
   c. The number of parameters must always be the same.
   d. The number of parameters must always be different.
4. (4 pts) What happens when you assign one object variable to another? For example:

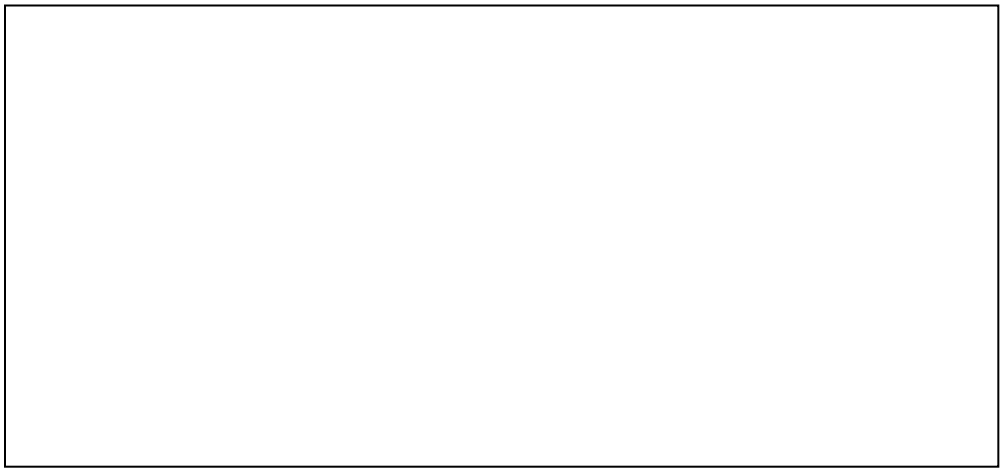
```
Die d1 = new Die();
Die d2 = d1;
```

   a. You get two copies of the same object.
   b. A compile-time error occurs.
   c. The object variables refer to the same object.
   d. It is illegal to assign one object variable to another.
5. (4 pts) In order to use an instance method, you must first do what?
   a. Construct an object.
   b. Design a public implementation.
   c. Create a static method.
   d. Declare a static object.
6. (4 pts) Static variables can be accessed from
   a. Static methods only.
   b. Non-static methods only.
   c. Static and non-static methods.
   d. The main method only.

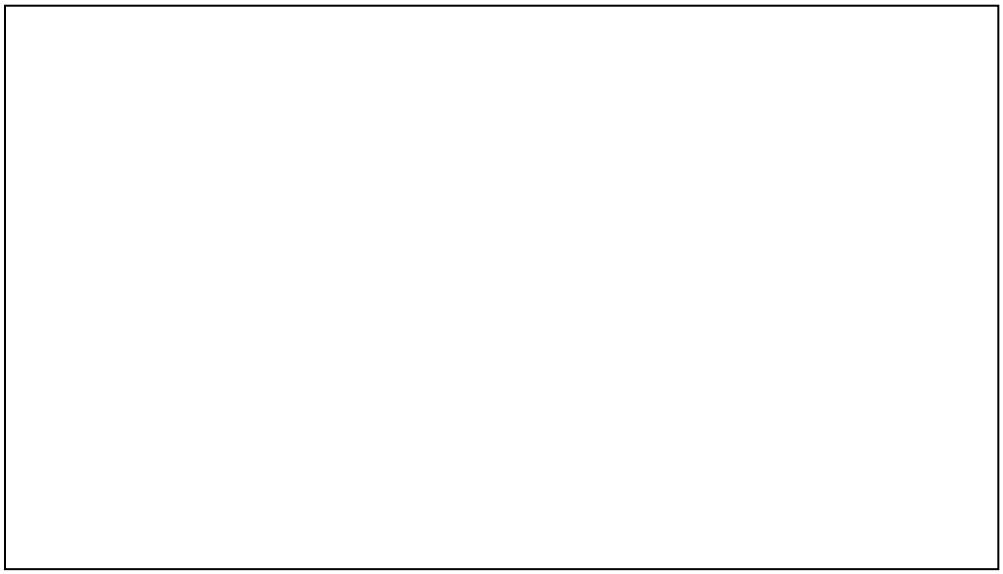
7. (8 pts each, 32 pts total) Starting with the skeleton code given below for a class representing a bank account, fill in the various sections as follows:
   a. **`Variables`**: Add two *instance* variables, one for the name of the account owner (String) and one for the balance (double). Also add a *class* variable (String) for the bank name, initialized to "Maps Credit Union".
   b. **`Constructor`**: The constructor should have two parameters whose values are used for setting the initial values of the two instance variables.
   c. **`Accessor`**: Add an accessor for the balance.
   d. **`toString`**: Add a toString method. If the owner's name were Kris Kringle and he had a balance of $2000, then the resulting `toString` should give:

      ```
      Maps Credit Union,  name: Kris Kringle,  balance: $2000.0
      ```

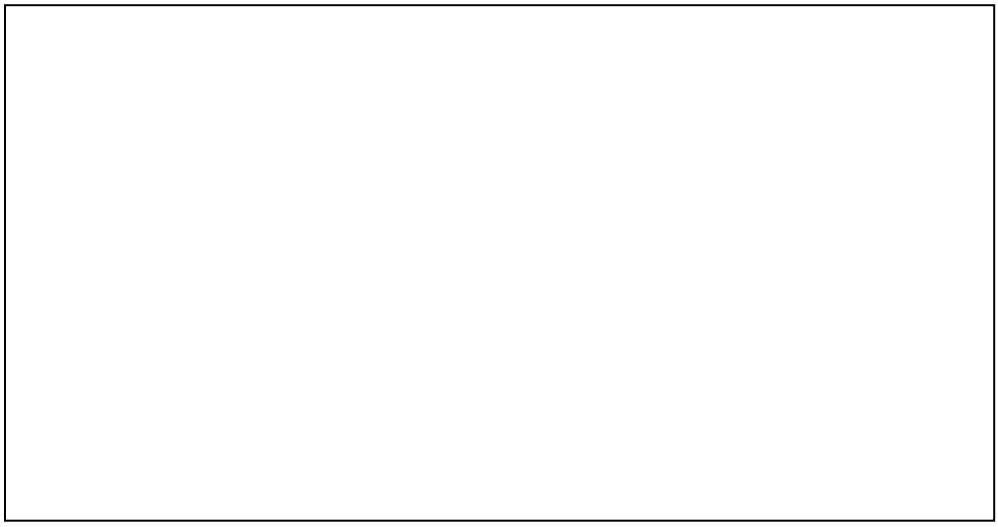
```
public class BankAcct {
      // Variables
```



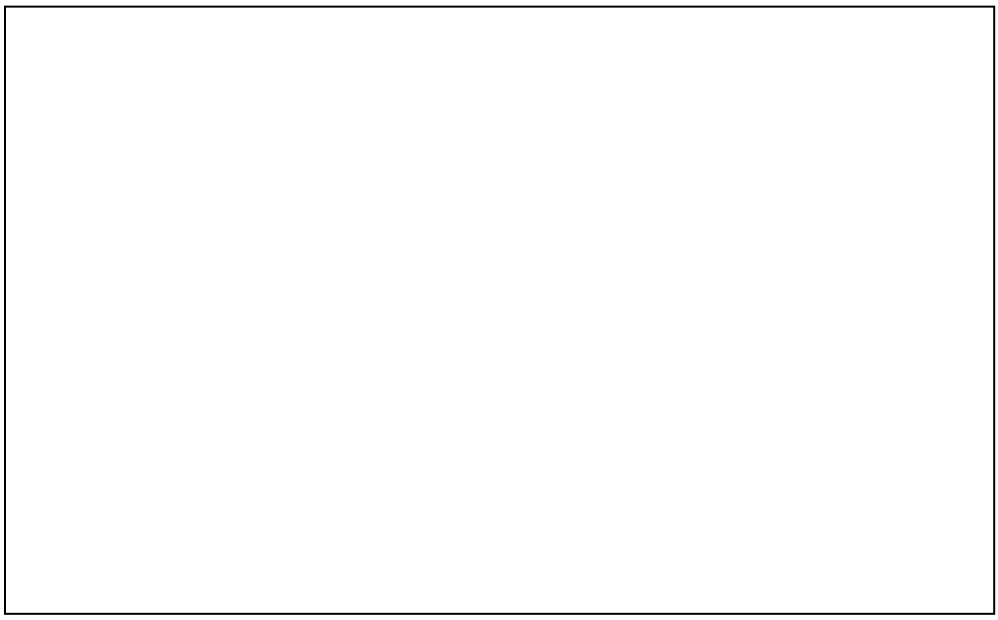
```
      //Constructor
```



```
//Accessor
```



```
//toString
```



```
}
```

8. (26 pts) Once we have the above class, we can write a second class to create several account objects. To the `CreateAccounts` class below, add code to do the following:

   a. **(**8 pts) **Santa:** Create a BankAcct object called `santa` with name Kris Kringle and a balance of $2000. Print `santa` using your toString method.

   b. **(**12 pts) **The Reindeer:** Kris is a generous guy, and happens to own a number of reindeer. He wants to make sure that each of his reindeer will have a little bit saved for an emergency. So, he opens up a bank account for each of them, each with a starting balance of $150.25. Create an ArrayList called `accounts` to store all of the reindeer accounts. Then, *using a loop*, add the reindeer accounts to your ArrayList. You will need to *index into* the `reindeerNames` array to obtain the name for each reindeer.

   c. **(**6 pts) Suppose Kris decides to change from the *Maps Credit Union* to his local *North Pole Credit Union*. To accommodate him, one can add a `changeBank` method to our `BankAcct` code (you don't need to write this for this exam). Then, in `CreateAccounts`, one can add the line in main

      ```
      santa.changeBank("North Pole Credit Union");
      ```

      Write any additional code (if any) that you would need to change all of the reindeer accounts to this same bank. (*Explain*).

```
import java.util.ArrayList;

public class CreateAccounts {

      public static void main(String[] args) {
         String[] reindeerNames = { "Dasher", "Dancer",
                          "Prancer","Vixen","Comet", "Cupid",
                          "Donner", "Blitzen", "Rudolph"};
```

```
// Santa
```

```
// The Reindeer
```

```
//Change banks
  santa.changeBank(“North Pole Credit Union”);
```

```
}
}
```

9. (3 pts each, 18 pts total) Based on the variables in the previous problem, match the following types:

   A. ArrayList of Strings
   B. ArrayList of BankAcct objects
   C. Array of Strings
   D. A BankAcct object
   E. `double`
   F. `String`
   G. `int`

   with each of the following:

i.	`accounts`	has type (circle one): A B C D E F G
ii.	`reindeerNames`	has type (circle one): A B C D E F G
iii.	`accounts.get(i)`	has type (circle one): A B C D E F G
iv.	`accounts.get(i).toString()`	has type (circle one): A B C D E F G
v.	`santa`	has type (circle one): A B C D E F G
vi.	`accounts.size()`	has type (circle one): A B C D E F G