Loop, Object, and String Exam A Name ___________________________

A. String method questions.
String x = “Friends and Family “; //the space at the end is a tab
String y = “Food to feed an army!”

1. (x.trim()).length() returns ___________
2. x.indexOf(‘i’, 3) returns ___________
3. (x + y).substring(12, 22) returns ___________
4. (y.toLowerCase()).charAt(0) returns __________
5. y.lastIndexOf(“feed”) returns __________
6. y.endsWith(“my!”) returns __________
7. x.compareTo(y) returns ___________ // indicate pos, neg, or zero number
8. y.replace(‘o’, ‘e’) returns _____________
9. x.indexOf(y.substring(12, 15)) returns ____________
10. x.charAt(x.length()) returns ____________
11. y.lastIndexOf(‘a’, 12) returns __________

B. String/Object “Mapping”

Consider the code fragment below:

```
String a = new String (“drumstick”);
String b;
String c = “thigh”;
String d = new String (a);
b = a;
a = c;
```

1. What do a, b, c, and d initially refer to, when they are first declared?
a = __________ b = __________ c = __________ d = _________

2. How many objects were created? _____

3. How many objects are accessible? _____

4. What do a, b, c, and d refer to at the end of the code?
a = __________ b = __________ c = __________ d = _________

5. At the end of the code, indicate if each of the following would return true or false.
a = = b returns __________ a = = c returns ___________
d = = a returns __________ a.equals(b) returns ___________
a.equals(c) returns ___________ d.equals(a) returns ___________

6. What are two other examples of objects we have used this year?

C. Loop with Switch

Trace the following loop, filling in the grid below. You may not need all of the space provided, or may need more. If the loop runs more than 5 times, just fill in the chart – you do not need to show more than 5 rows.

```
byte b = 3;
for (int x = 15; x<35; x + = b){
    b = (byte)(x/2);
    switch(x){
            case 12: case 13: case 15:
                    System.out.print(“Corn”);
            case 16: case 20: case 21:
                    System.out.print(“Bread”);
                    break;
            case 22:
                    System.out.println(“Potatoes”);
                    break;
            case 34:
                    System.out.print(“Jello”);
            default:
                    System.out.println(“Pie”);
    }
System.out.print(“,”);
}
```

Loop	x	b	Output
1			
2			
3			
4			
5			

D. Program

Leaves receive light and use this energy to create sugars, in the process known as photosynthesis. It takes two photons of light for a single molecule of glucose (a sugar) to be made.

Write a program that:

- Receives input of the genus and species of the tree and stores it as the String treeName (which has already been declared).
- Receives input of light/dark data for 10 seconds, which is recorded as a String called lightDarkData (it has already been declared).

The light/dark data will be a String of ten ‘l’ or ‘d’ characters to indicate the amount of light the tree is receiving (a scientist can measure this amount with a probe similar to a solar cell).

- Output the shortened scientific name for the plant, by printing the first letter of the genus, capitalized, followed by a period, then the species name in all lowercase letters.
- Output how many molecules of glucose can be made in a minute, by counting up the ‘l’s from lightDarkData and multiplying by 3. Use a for loop to count these up.
- Only use String methods taught in class to process the input and produce the output.
- Write the program so it will continue until “quit” is entered, using a do-while loop. Check for any version of “quit,” with capital or lowercase letters in any combination.
- **Fill on the back of this page to complete the program.**

Example:

```
Enter the tree’s genus and species:  Acer saccharum
Enter light/dark data for 10 seconds: ldldllllldl

This A. saccharum can make 21 molecules of glucose per minute.

Would you like to enter a new tree?  Type “quit” to quit, or anything else to continue:  go

Enter the tree’s genus and species:  sequoia GiGantea
Enter light/dark data for 10 seconds: ldlddldddl

This S. gigantea can make 12 molecules of glucose per minute.

Would you like to enter a new tree?  Type “quit” to quit, or anything else to continue: qUiT
```

```
public class Photosynthesis{

        String treeName;
        String lightDarkData;


                                                // you will need more variables.  Declare them here or in main()

        public static void main (String [] args){
                TextReader tr = new TextReader();
                do{
                                                                       //get name and data input





                                                                      //process treeName to produce output


                        for (                                                           ){//for loop - count up 'l's






                        }                                              // end of loop
                                                        //calculate glucose by multiplying number of 'l's by 3


                                                                //print output


                                                             //prompt to see if user wants to quit or continue


                        }while( ____________________________);   //check for sentinel value
        }
}
```