

1) (30 points; 5 points each) Write the output of each piece of code. If the code gives an error, write any output that would happen before the error, and then write "ERROR". If there is an infinite loop, write what that output is the first three times through the loop, and then write "INFINITE".

	Code	Output
a)	<pre>int a = 5; int b = 2; double c = 5; System.out.println(((double) a)/b); System.out.println(a/b); System.out.println(c/b); System.out.println(b/a);</pre>	
b)	<pre>String s = "Lehman"; int i = 1; while (i < s.length()) { System.out.println(s); s = s.substring(i); i++; } System.out.println(s);</pre>	
c)	<pre>for (int i = 0; i < 4; i++) { for (int j = i; j < 4; j++) { System.out.print(j); } System.out.println(); }</pre>	
d)	<pre>boolean y = true; int k = 0; while (y) { k++; System.out.println(k); } System.out.println(y);</pre>	

1) (Continued)

	Code	Output
e)	<pre> public class Abc { int a = 0; int b = 0; public Abc(int c) { a = c; } public String toString() { return "a=" + a + "; b=" + b; } public static void main(String[] args){ Abc f = new Abc(5); System.out.println(f); Abc g = f; System.out.println(f); System.out.println(g); g.b = 7; System.out.println(f); System.out.println(g); } } </pre>	
f)	<pre> public class Test { static int num = 0; String s = "cat"; public Test() { num++; } public void fn() { s = "" + s.charAt(0); num = 44; } public static void main(String[] args) { Test a = new Test(); System.out.println("a:" + num+ " " +a.s); Test b = new Test(); System.out.println("b:" + num+ " " +b.s); a.fn(); System.out.println("a:" + num+ " " +a.s); System.out.println("b:" + num+ " " +b.s); } } </pre>	

2) (30 points) The Java class called `BaseballTeam` is started below. An object of class `BaseballTeam` represents a Major or Minor League Baseball team. This class has four private instance variables:

- `name`, which is a `String` representing the name of the team
- `numGamesPlayed`, which is an `int` representing the number of games played so far in this season
- `numGamesWon`, which is an `int` representing the number of games won so far this season
- `isMajor`, which is a `boolean` representing whether the team is a major or minor league team (set to `true` for major, and `false` for minor)

```
public class BaseballTeam {
    private String name;
    private int numGamesPlayed;
    private int numGamesWon;
    private boolean isMajor;

    // your code will go here

}
```

a) (7 points) Write a constructor which takes in a name, number of games played, number of games won, and boolean of whether the team is a major league team or not, and assigns those values to a `BaseballTeam` object.

b) (5 points) Use your constructor from part (a) to instantiate an object of class `BaseballTeam` with name "New York Yankees", 162 games played, 87 wins, and a major league team.

2) (continued)

c) (8 points) Write a setter (mutator) method for the instance variable `numGamesPlayed`. This setter should only allow `numGamesPlayed` to be set to 0 or a positive number. If a negative number is passed in, a error message should be printed and the value of `numGamesPlayer` should not be changed.

d) (10 points) Write a non-static method `isWinningTeam` which does not have a parameter, and returns a `boolean` value. This method should return `true` if the team has won half or more of the games played, and `false` if not.

3) (10 points) Write a method `rearrange` which does **not** return anything and has one parameter: a `double` array. This method should:

- replace the first element of the array with the last element of the array
- replace all other array elements with the original first element of the array

If the array has length 0 or 1, do not change the array. For example, if the array is

4.2	-10.0	9.5	0.0	2.66
-----	-------	-----	-----	------

then `rearrange` should change it to

2.66	4.2	4.2	4.2	4.2
------	-----	-----	-----	-----

4) (15 points) Write a piece of code which allows the user to enter the grades for multiple tests and average them. Specifically, this code should:

1. ask the user how many tests they have, and how many questions each test has. (You may assume all tests have the same number of questions.)
2. ask the user for the scores for each question for each test.
3. print the total score for each test (the total score is the sum of the scores for all questions in that test)
4. print the average total score (averaged over all the tests)

6) (15 points) Consider the Java GUI below that contains 1 TextField, 2 Labels, and 2 Buttons:

```
public class Counter extends JFrame implements ActionListener{
    JTextField text1 = new JTextField(20);
    JButton countButton = new JButton("COUNT");
    JButton anotherButton = new JButton("DOES NOTHING");
    JLabel label1 = new JLabel("Enter a String:");
    JLabel label2 = new JLabel("Number of a's:");

    // other methods in the class, including the
    // constructor and main method are not shown here

    public void actionPerformed(ActionEvent e){
        // your code will go here
    }
}
```

Define the body of the `actionPerformed` method as follows:

- if the button labeled `COUNT` is clicked, then `JLabel label2` is set to be the number of letter `a`'s in the `String` entered in `JTextField text1`

Remember that there is NO `count` method in Java. You may define and use a helper method.