Name _______________________________________________________________

Question One. 25 points. Consider the supplemental information provided. Complete the printChain() method in the MonkeyChain class using a **standard for loop**. (You can earn 15 of these points if you don’t use a standard for loop.) No comments are necessary.
Question Two. 25 points. Consider the supplemental information. Complete the longestRun() method in the MonkeyChain class using a **for each loop**. (You can earn 15 of these points if you don’t use a for each loop.) No comments are necessary.
Question Three. 25 points. Consider the supplemental information provided. Complete the countRedMonkeysAux() method in the MonkeyChain class using recursion. (You can earn 15 of these points if you solve the problem non-recursively.) No comments are necessary.
Question Four. 25 points. Rewrite the getContents method below to incorporate exception handling. In particular, when an ArrayIndexOutOfBoundsException is generated, the method should return the value -1.0. No comments are necessary.

```java
public class QuestionFour {
    private double [] numbers = {1.0, 2.0, 3.0, 4.0};

    public double getContents(int index) {
        return numbers[index];
    }
}
```
Driver.java

public class Driver{
    public static void main (String [] args){
        MonkeyChain monkeys = new MonkeyChain(); // creates an initial chain with one red monkey
        process(monkeys);
        for (int i = 1; i <= 2; i++) // add two green monkeys
        {
            monkeys.addMonkey("green");
            process(monkeys);
        }
        for (int i = 1; i <= 3; i++) // add three red monkeys
        {
            monkeys.addMonkey("red");
            process(monkeys);
        }
    }
    private static void process(MonkeyChain chain){
    System.out.println("Monkey chain length = " + chain.chainLength());
    chain.printChain();
    System.out.println("The number of red monkeys = " + chain.countRedMonkeys());
    System.out.println("The largest run with the same color = " + chain.longestRun() + "\n");
    }
}

MonkeyChain.java

import java.util.ArrayList;

public class MonkeyChain{
    private ArrayList<String> monkeyChain;
    MonkeyChain(){
        monkeyChain = new ArrayList<String>();
        addMonkey("red");
    }
public void addMonkey (String color)
{
    monkeyChain.add(color);
}

public int chainLength()
{
    return monkeyChain.size();
}

public int countRedMonkeys()
{
    return countRedMonkeysAux(0);  // start counting with the contents of slot 0
}

Output Produced

Monkey chain length = 1
Monkeys from top to bottom: red
The number of red monkeys = 1
The largest run with the same color = 1

Monkey chain length = 2
Monkeys from top to bottom: red -> green
The number of red monkeys = 1
The largest run with the same color = 1

Monkey chain length = 3
Monkeys from top to bottom: red -> green -> green
The number of red monkeys = 1
The largest run with the same color = 2

Monkey chain length = 4
Monkeys from top to bottom: red -> green -> green -> red
The number of red monkeys = 2
The largest run with the same color = 2

Monkey chain length = 5
Monkeys from top to bottom: red -> green -> green -> red -> red
The number of red monkeys = 3
The largest run with the same color = 2

Monkey chain length = 6
Monkeys from top to bottom: red -> green -> green -> red -> red -> red
The number of red monkeys = 4
The largest run with the same color = 3