CSCI 203: Introduction to Computer Science I
Bucknell University
Computer Science Department
http://www.eg.bucknell.edu/~csci203

Project 5

October 21, 2009

Table of Contents

1. Problem Description

In this project you are writing code that will be used to analyze test data for wireless transmitters. You are to check two measurements:

- The power output of the transmitter must be within 90% and 110% of the specified level. You will be given an *output power ratio*, and you will test that the value is greater than or equal to 0.9 and less than or equal to 1.1.
- The bit rate of the transmitter must be greater than 11 Mb/second. You will be given a *bit rate* in Mb/sec; you will test that it is greater than 11.

Your program should allow a test technician to input the two values for the test parameters, should check the test values against the specified values, and should print out a report. The report should display the input values and state if the transmitter passes the test or not. If the transmitter fails the test, your program should print out the reason(s) why the transmitter failed.

You may use either the `Scanner` class or dialog windows for input.

2. Details

You will need two classes to complete this project. This section specifies those classes:

2.1. Transmitter Class

A `Transmitter` object holds the two pieces of test data for a transmitter: a `double` for the power output ratio, and an `int` for the bit rate.

This class should have a default constructor (think about reasonable values to use as defaults for your data), a two-parameter contructor that has parameters to hold the test values, which are used to set the two instance fields, and a `toString` method for printing the test data being processed. Here's an example of what `toString` might produce:

```
Power output ratio:  1.30; bit rate:  21
```

Hint: You've seen methods in earlier labs that you can use in implementing your `toString` method. You should also consider the static `format` method of the `String` class, which works much like the `printf` method. See the API for details.

You are not required to write accessor methods for the instance fields.

You should also write two methods that analyze the test results:

- a method `isOutputWithinRange` that returns a `boolean` value; the method should return `true` if the output power ratio is in the specified ranges and `false` otherwise.
- a method `isBitRateSufficient` that returns a `boolean` value; the method should return `true` if the bit rate is high enough and `false` otherwise.

Be sure to avoid magic numbers in implementing these methods!

Finally, you should also write a `main` method that tests the constructors and all other methods. Make sure to state clearly in a `System.out.println` statement what you are testing in each case.

2.2. TestTransmitter Class

The `TestTransmitter` class should have a `main` method that prompts the user for the two test values for a transmitter and reads the input values. You may assume that the input is

of the proper type; you do not need to handle cases where the wrong type of information is input.

The `main` method should use the test data to construct a `Transmitter` object. The `main` method should then use the `Transmitter` methods to determine if the transmitter passes the tests and to format a report. The report should contain the following information:

- A summary of the test data printed using the `toString` method of the `Transmitter` class (use the specified method; do not use accessors and format the output in the `main` method).
- A statement either that the transmitter passed **both** tests (note in this case we don't want to see individual results), or a statement of which test (or tests if it fails both) the transmitter failed.

Here's an example of a test report where all tests were passed (note that only the overall result is reported rather than individual test results):

```
Testing transmitter: Power output ratio:  1.05; bit rate:
      14
Transmitter passes all tests.
```

Here's an example where one test was failed:

```
Testing transmitter: Power output ratio:  1.15; bit rate:
      23
Transmitter fails power output test.
```

Here's an example where both tests were failed:

```
Testing transmitter: Power output ratio:  0.50; bit rate:
      10
Transmitter fails power output test.
Transmitter fails bit rate test.
```

3. What to Submit

Before submitting your project, check to be sure that you have Javadoc comments for each class, constructor, and method.

Your project folder must be named `proj5-xyz001` where `xyz001` is your login name. Drag your project folder into the drop box for your lecture instructor.