CS 122 – Lab #4 – I/O Practice

Today's lab will feature two exercises to practice some of the I/O concepts you have seen lately. In particular, you will practice some file I/O, formatted output, tokenizing strings, and redirecting standard I/O at the shell.

On your cs.furman.edu account, create a new directory called `lab04` and do all of today's work in there. Copy the files `~chealy/www/cs122/lab04/*.txt` to your directory. These are example input files for the lab.

Program #1: `WinLoss.java`

This program will read from standard input. The input will contain win/loss records of an athletic team. Each line of input will show the record of one season. Your job is to compute the team's overall record. On each line of input, there will be two numbers separated by a hyphen, with no whitespace. For example:

```
17-3
1-14
5-6
8-0
12-10
```

Do not assume a certain number of digits for each number. Use `StringTokenizer` to help you grab the number of wins and losses from a line of input. Add up the total number of wins and losses, and print these numbers out using `System.out.printf()`. Also, indicate the overall percentage of games won, rounded and displayed to the nearest tenth of a percent. Don't forget to print the percent sign. Since the program will not be prompting the user for input, these results should be the only output from the program.

How do we run a program like this? There are two ways. One way is to run the program interactively. This means the user will type a few lines of data, and then hit CTRL-D to signal to the program that there is no more standard input. But this method of typing all the input is rather tedious in the long run. So, the second way we can run this program is to redirect standard input from a file. Try this command:

```
java WinLoss < record.txt
```

Program #2: `Length.java`

The file `movies.txt` shows a list of movie titles along with their respective lengths. The purpose of this program is to calculate the total length of all these films.

Since this program will explicitly perform file input, your will need to import `FileInputStream`.

For each line of the input file, you should tokenize the hyphen, in order to separate the title of the movie from its length. We will discard the movie title. You just need the length. However, the length is itself a string, of the form `hh:mm:ss`. This length string must itself be tokenized! You should declare a second `StringTokenizer` object to handle the colons inside this string.

To handle the arithmetic of this problem, here is one possible approach. As you read the input, only keep track of the total number of seconds. Use the fact that there are 3600 seconds in an hour and 60 seconds in a minute to compute the number of seconds of each movie. At the end of your program, you should convert this total number of seconds back into `hh:mm:ss` format, and display this final answer.

For your convenience, the file `movies.txt` is reproduced here:

```
3:10 to Yuma - 1:32:00
Back to the Future - 1:56:04
The Bishop's Wife - 1:49:29
The Day the Earth Stood Still - 1:32:07
E.T. - 2:00:03
Home Alone - 1:42:50
The Lord of the Rings:  The Fellowship of the Ring - 3:48:12
Miracle - 2:15:55
My Man Godfrey - 1:33:18
Oklahoma - 2:20:02
Singin’ in the Rain – 1:42:46
Some Like It Hot - 2:01:35
Tora Tora Tora - 2:24:46
What's Up Doc - 1:33:50
```