Reading and Reference

- Glass and Ables, Chapters 4, 5, and 8
- Classic Shell Scripting book (course web site)
- Mac OS X Developer Library (course web site)

Exercise 1. In class, we developed a shell script named `sum` which takes one command-line argument, N. In response, it validates the input and computes the sum of the first N positive integers. Modify this script to obtain the following:

- One command-line argument can appear, with the behavior described above.
- Two command-line arguments, L and U, can appear. In this case the sum of all integers starting at L and ending at U is computed. L and U may be negative, provided $L \leq U$. For example, the invocation

  ```
  sum -3 2
  ```

  denotes the sum $(-3) + (-2) + (-1) + 0 + 1 + 2$.
- The modified script should continue to check for error conditions and give appropriate messages and exit codes.

Exercise 2. Implement a shell script, named `ex2`, which will provide a listing of all those files in the current working directory whose size (in bytes) is greater than a given threshold, given as a command-line argument. For example,

```
ex2 20000
```

should give a listing of all files in the current directory larger than 20,000 bytes. Hint: there is a very powerful Unix utility by the name of `find`. As a starting point, read the manual page for this command. Your script should include appropriate error checks.

Exercise 3. Implement a shell script, named `ex3`, which has one command-line argument — which is a filename extension, such as `tex`, `c`, or `pdf`, for example. This script should make a copy of all of the files with this extension, appending `.copy` to the name. For example,

```
ex3 tex
```

will make copies of all `.tex` files in the current directory, adding `.copy` to its original name. For example, if you had files `hw01.tex` and `hw02.tex` in your directory, these would be copied to `hw01.tex.copy` and `hw02.tex.copy`.

Exercise 4. Implement a shell script, named `ex4`, which has an arbitrary number of command-line arguments. For each argument, your script should report the number of lines, words, and bytes in the file. For example,

```
ex4 hw01.tex hw02.tex
```

would produce output in the following format:

```
hw01.tex has 364 lines, 1498 words, and 12317 bytes.
hw02.tex has 286 lines, 1331 words, and 10819 bytes.
```

Of course, wild card characters could appear as command line arguments, such as:

```
ex4 *.tex
```

Your script should include appropriate error checks. Hint: read the manual pages for the `wc` and `cut` utilities. Some experimentation at the command-line prompt may be helpful.

Exercise 5. Implement a shell script, named `ex5`, which outputs the total number of bytes in a named directory. For example,

```
ex5 $HOME
```

might produce a value such as

```
5697536
```

which is the size, in bytes, of the home directory.

What to Submit

Put copies of your completed scripts into a folder, then drag this folder onto the EIU submit icon. Be sure that your name appears in each script as a comment somewhere near the top of the file.