ITEC 2610 Object-Oriented Programming
Sections A/B/C, Fall 2019

Assignment 3

To submit:

Prepare a single `.zip` file containing all the source files (**.java files, NOT .class files**).
Submit this file through Moodle.

```
/******************************************************************

Marking Template:


Style (variable naming, indentation, & layout)  _____/5

Comments                                        _____/5

Proper error/exception handling

        (by inspection)                         _____/15

Code Compiles?                                  _____(yes/no)

Successful execution of test cases              _____/75

Total                                           _____/100


According to this template the maximum mark you can get for code that does

not compile is 25/100.

******************************************************************/
```

Details

Write a program `NumberCount` that counts the numbers (including integers and floating point values) in one or more text files. Note that only numbers separated by whitespace characters are counted, i.e., only those numbers that can be read by either `readInt()` or `readDouble()` are considered.

Start a new thread for each file. The last active thread also prints a combined count. Use locks to protect the combined count and a counter of active threads. For example, if you call

```
java NumberCount report.txt address.txt Homework.java
```

then the program might print (assuming those files exist in the current directory)

```
address.txt: 1052
Homework.java: 445
report.txt: 2099
combined count: 3596
```

Errors/exceptions should be handled properly:

1. When the command line arguments do not follow the given format (e.g., if you run "java NumerCount" without any filenames provided), then the program should print a help message telling the correct way to use the program.

2. When one or more files specified as the command line arguments do not exist, the program will print out a message saying that the file(s) do not exist; in this case, the program will still print the given filename, and a count 0.

3. Other types of I/O exceptions can be handled by printing out a generic error message. If there is an exception during the reading of a file, the count to be printed out for that file is whatever the count is up until the exception.