# CPSC 470 – Design Patterns – Homework #1

This assignment has two parts, but you will only need to turn in the end product, not the intermediate step between parts 1 and 2.

Problem Statement: Upon arriving at your new job working for Google, they decide to haze the “newbie” by giving you the ugliest program in the world to fix. You are given the program that contains 3 different parts that cause the world to rotate on its axis, birds to sing, and bees to buzz. The code currently contains several print statements in each class.

***Task 0:*** Get the ugly code to execute as-is in your development environment.

***Task 1:*** You are required to convert all the various print statements into log statements instead. Create a class called Logger that has a single method with the signature:

public void log(Object originator, String text);

Each time the log method is called, you will print the output to the console as follows:

s.ss – originator class: text

where originator class is given by originator.getClass().getName() and text is the text from the print statement.

Your times (s.ss) should be the number of seconds since the program began (or at least since the logger was first instantiated). Use the singleton pattern to ensure that all of your classes are using the same logger.

***Task 2:*** The team would like you add the option to write the log file to a file instead of the console. Without getting too fancy and worrying about overwriting log files, you will modify your logger code to add the option in the getInstance() method to indicate a boolean parameter. If the boolean isConsole is true, then you continue to write to the console. Otherwise, you will create a file called microsoftSucks.log (this is Google after all) and write all of your log statements to that file instead of the console.

Use a Factory Method pattern to choose which logger to instantiate. This should probably cause you to have 3 Logger classes: Logger, ConsoleLogger, and FileLogger.

## To Submit:

Upload a zip file containing:

* All of the code, both yours and the original classes (as modified by you), in the proper folder/package (hw1 in this case)
* A class diagram of the finished program. This can be hand drawn and then photographed if you want.