```java
/* Message Levels:
 * UNIMPL
 * VERBOSE  - Way to much detail
 * INFO     - Normal running, useful to follow execution
 * WARNING  - Something wierd is going on, someone fucked up
 * RED      - Recoverable error (one query failing, one timeout)
 * ERROR    - Something went badly wrong
 * FATAL    - Going to crash, far more worrying if it doesn't crash
 * CRITICAL - Fuck everything, the moon is purple
*/

package ballmerpeak.turtlenet.server;

import java.io.*;
import java.util.Date;

public class Logger {
    static boolean started = false;
    static String path;
    static PrintWriter log;

    public static void init (String logfile) {
        if (!started) {
            started = true;
            path = logfile;

            try {
                log = new PrintWriter(new BufferedWriter(new FileWriter(path)));
                log.println("===== Turtlenet started at " + new Date() + "=====");
                log.flush();
            } catch (Exception e) {
                throw new RuntimeException("ERROR: Unable to open log: " + e);
            }
        }
    }

    public static void close () {
        if(started) {
            log.println("===== Turtlenet closed  at " + new Date() + "=====");
            log.flush();
            log.close();
        }
    }

    public static void write (String level, String place, String s) {
        if (started) {
            log.println((System.currentTimeMillis()/1000L) + " " + level + getTabs(level) + place + "\t" + s);
            log.flush(); //In case of a crash we don't want to be digging up the wrong code
        }
    }

    private static String getTabs (String s) {
        s = (System.currentTimeMillis()/1000L) + " " + s;
        if (s.length() < 16)  return "\t\t"; else return "\t";
    }
}
```