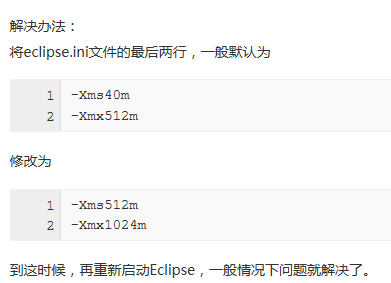
java.lang.OutOfMemoryError: GC overhead limit exceeded

分析：异常分析



在这个时候有几个问题需要思考一下：

•为什么会出现这个错误

•Xms和Xmx是什么意思(我到底修改了什么)

为什么会出现这个错误或者说这个错误是什么意思？

也就是说GC试图回收内存，但是什么也没有回收到。默认情况下，JVM花费了98%的时间在GC上，但是GC过之后只有不到2%的堆内存被回收。

这里边比较有意思的是98%指的是时间，而2%指的却是堆内存。

到这里可以再拓展思考一下，如果没有这个GC overhead limit会发生什么现象

**如果没有这个GC overhead limit会发生什么现象？**

简单来讲，就是一次GC过后，并没有回收到内存，很快又会进行GC，极端情况下会循环GC(关键是并没有回收到可用内存)，从而导致CPU 100%负载。

### Xms和Xmx是什么意思(我到底修改了什么)

这两个都是JVM的参数

* Xms JVM的初始堆大小
* Xmx JVM的最大堆大小

当**最小堆**被占满后，会进行**GC**，如果GC之后还**不能得到足够的内存**，就会扩展堆，最大扩展至**Xmx设定的[1]值**。

**一、异常如下：**Exception in thread "main" java.lang.OutOfMemoryError: GC overhead limit exceeded

**二、解释：**  
JDK6新增错误类型。当GC为释放很小空间占用大量时间时抛出。  
一般是因为堆太小。导致异常的原因：没有足够的内存。

**三、解决方案：**

1、查看系统是否有使用大内存的代码或死循环。  
2、可以添加JVM的启动参数来限制使用内存：-XX:-UseGCOverheadLimit