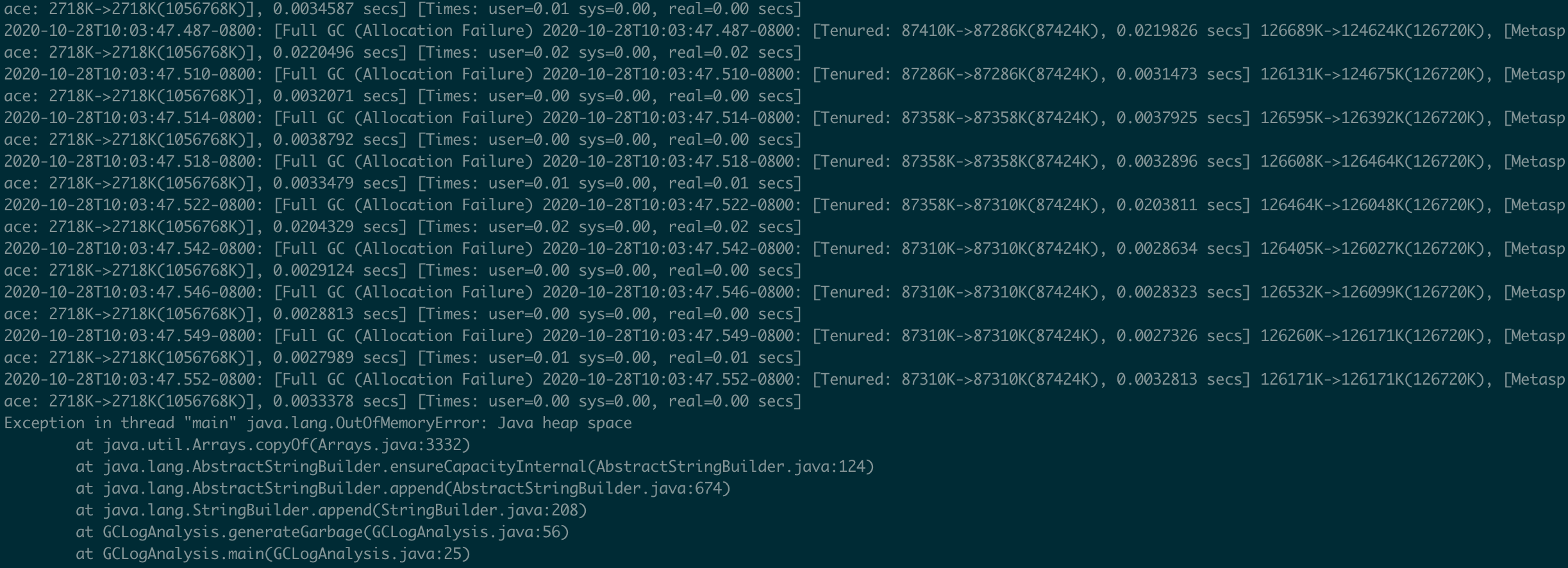
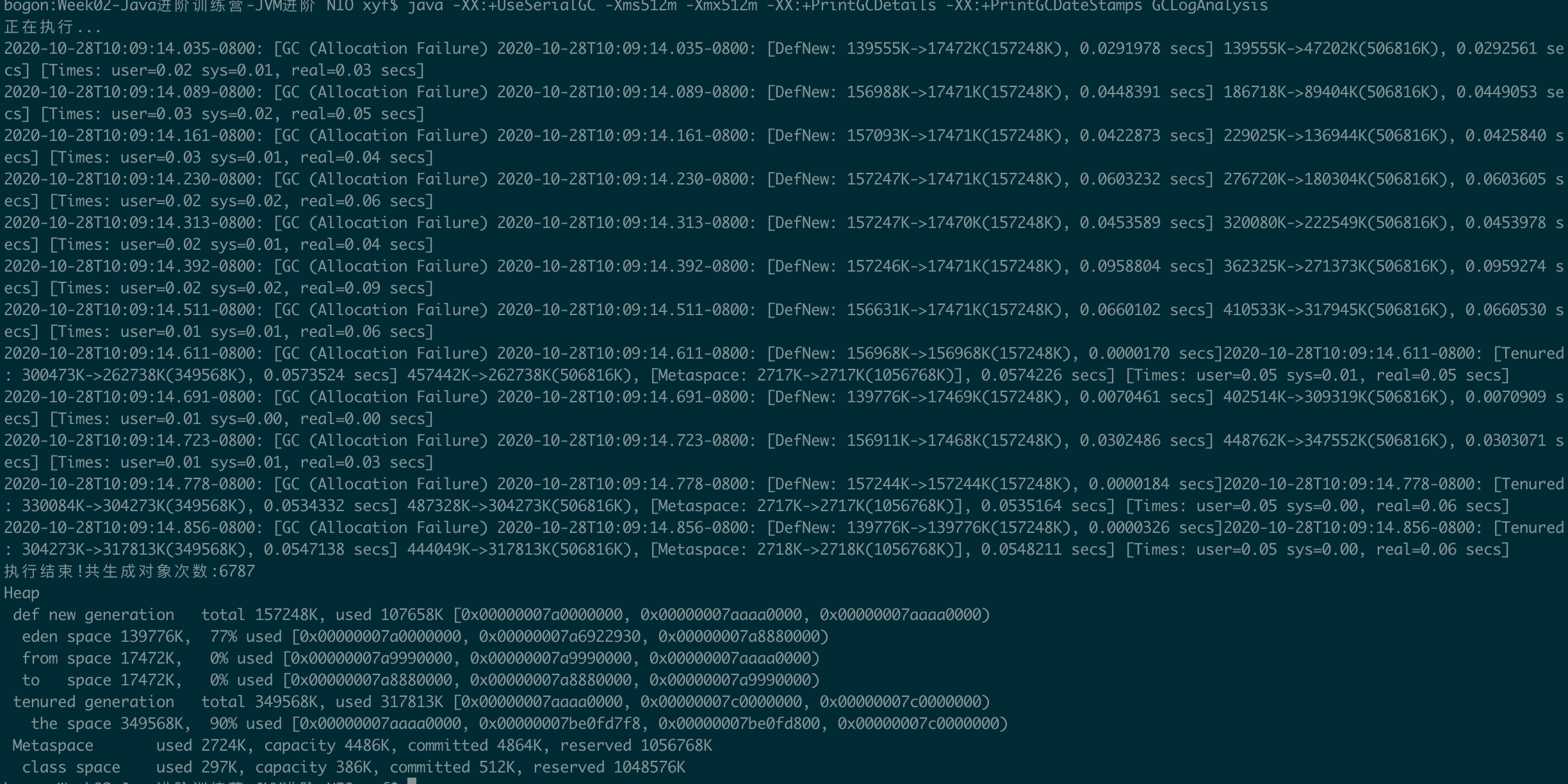
作业1：

（一）执行结果记录

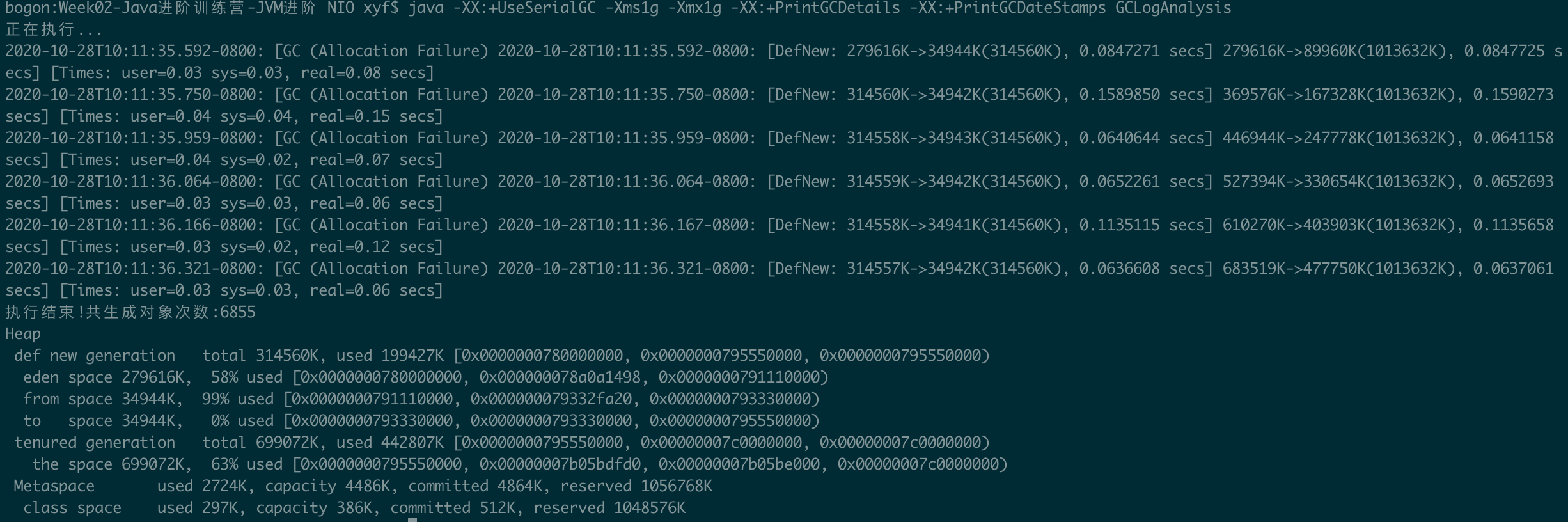
1. 串行GC
   1. -XX:+UseSerialGC -Xms128m -Xmx128m



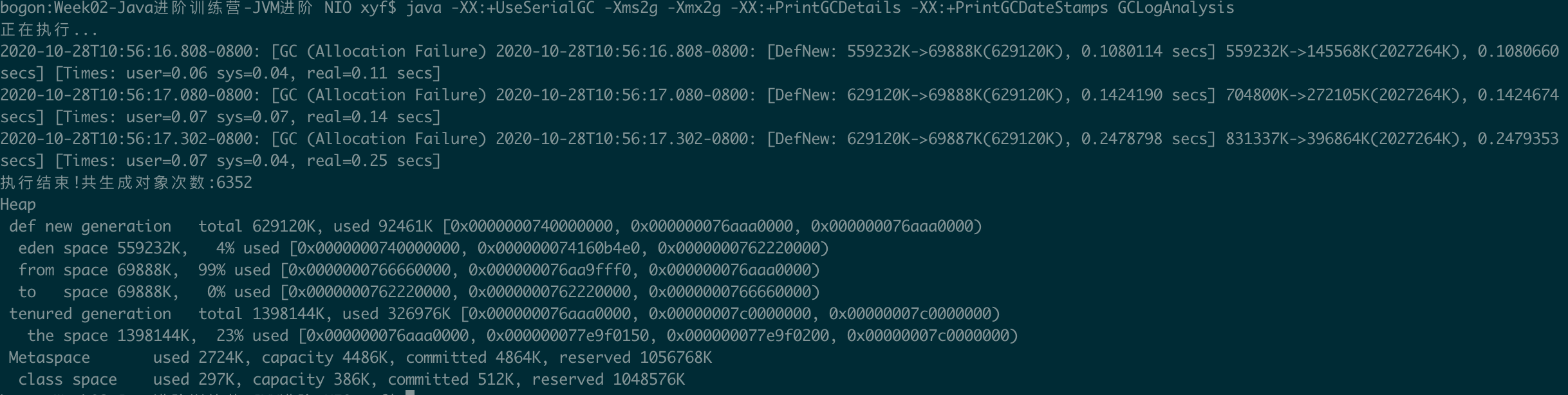
* 1. -XX:+UseSerialGC -Xms512m -Xmx512m



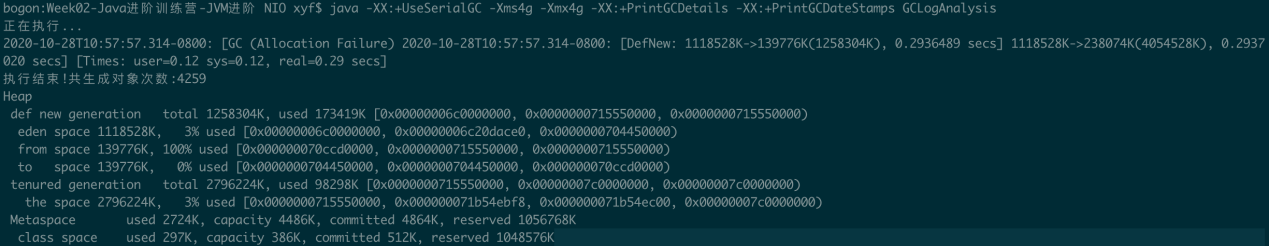
* 1. -XX:+UseSerialGC -Xms1g -Xmx1g



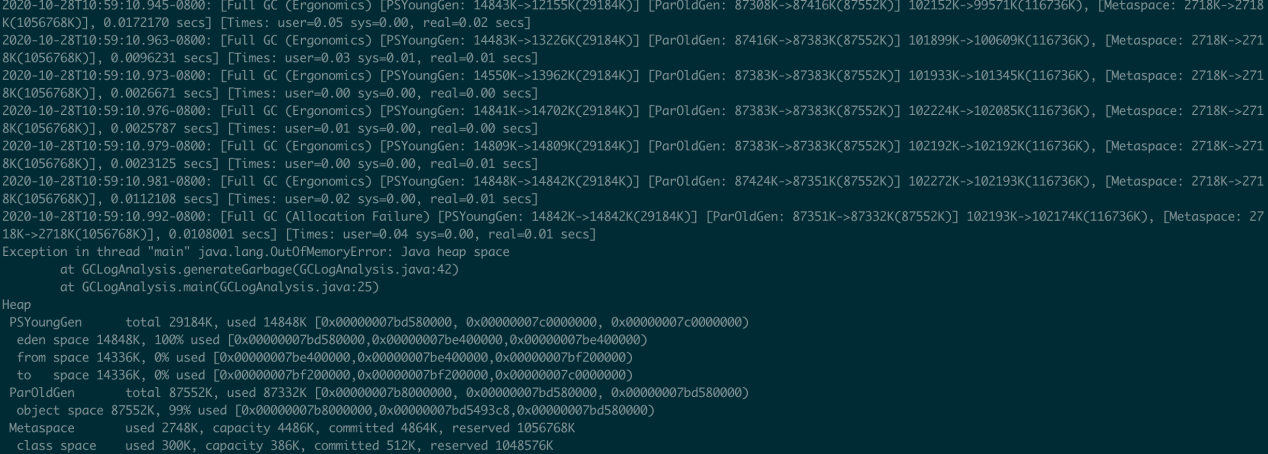
* 1. -XX:+UseSerialGC -Xms2g -Xmx2g



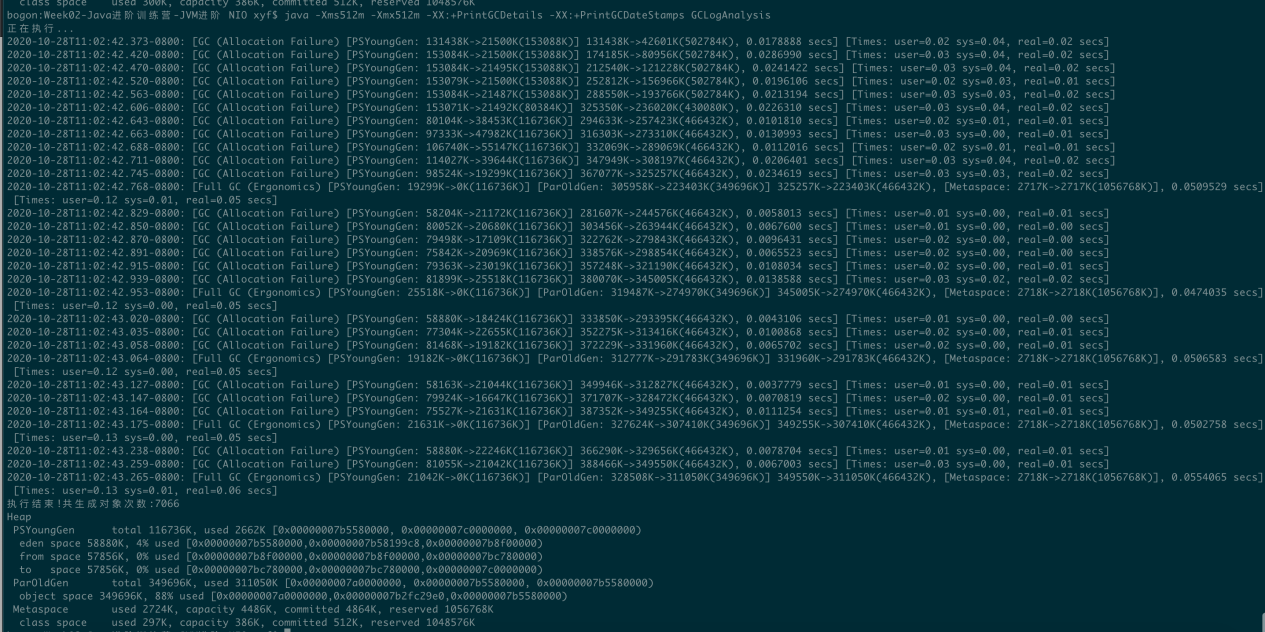
* 1. -XX:+UseSerialGC -Xms4g -Xmx4g



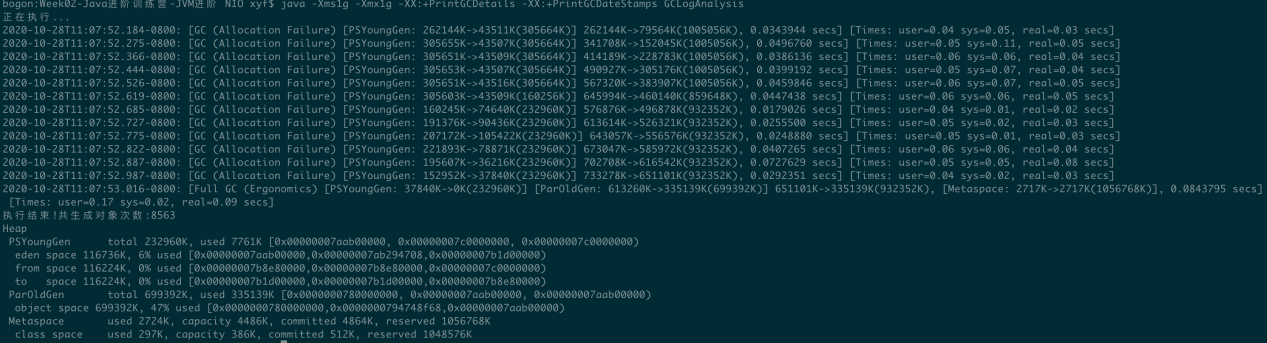
1. 并行GC
   1. -XX:+UsePallelGC -Xms128m -Xmx128m



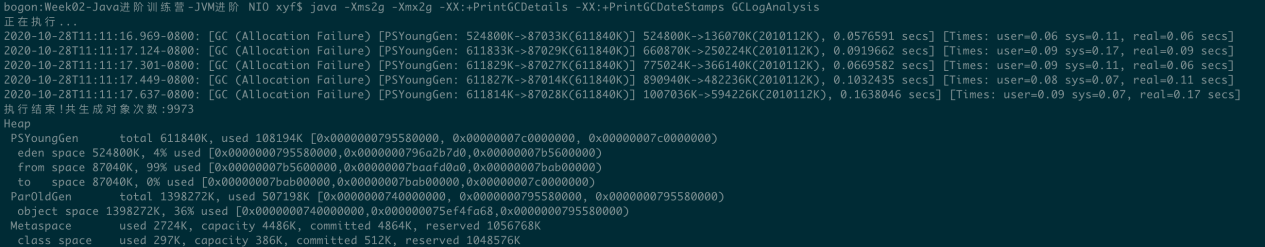
* 1. -XX:+UsePallelGC -Xms512m -Xmx512m



* 1. -XX:+UsePallelGC -Xms1g -Xmx1g



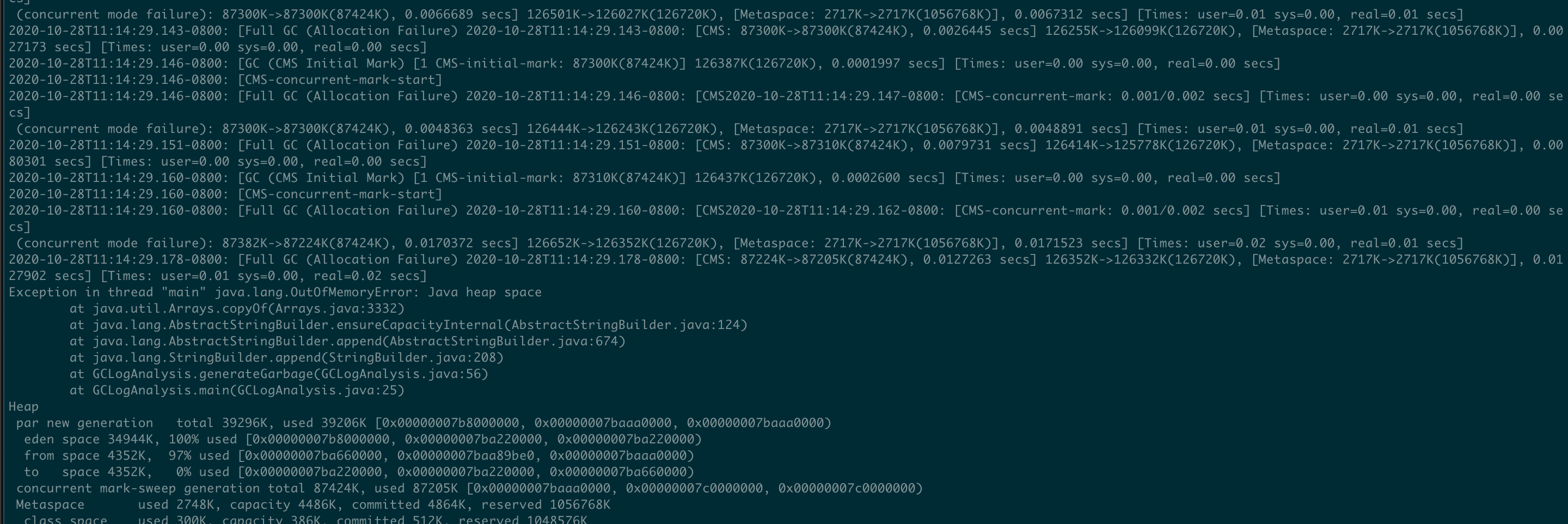
* 1. -XX:+UsePallelGC -Xms2g -Xmx2g



* 1. -XX:+UsePallelGC -Xms4g -Xmx4g



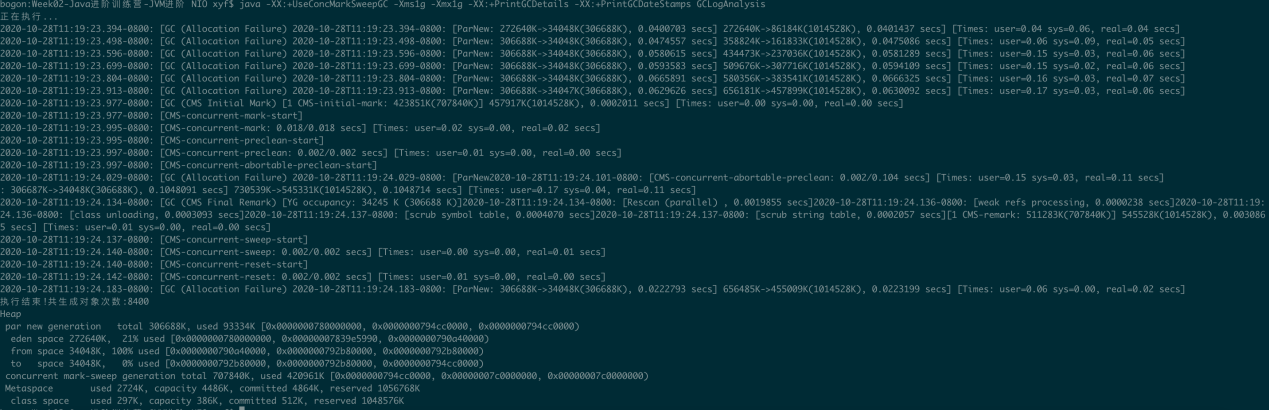
1. CMS GC
   1. -XX:+UseConcMarkSweepGC -Xms128m -Xmx128m



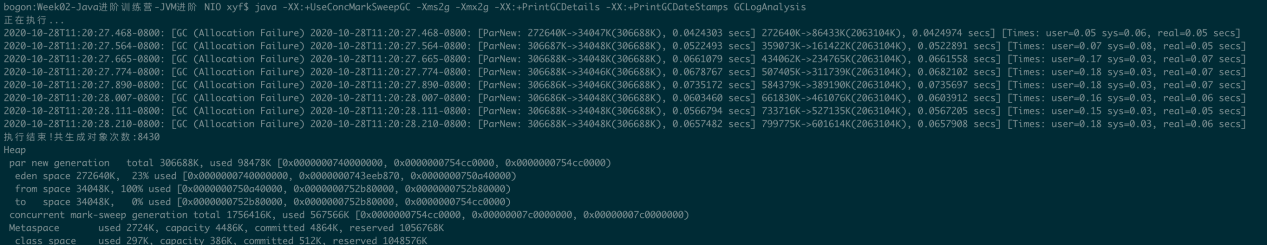
* 1. -XX:+UseConcMarkSweepGC -Xms512m -Xmx512m



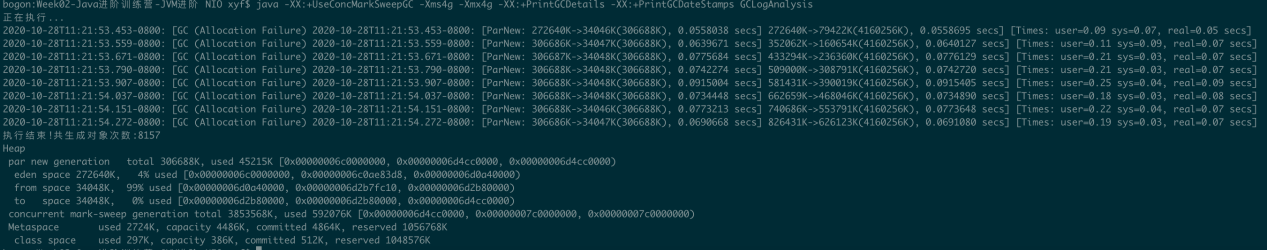
* 1. -XX:+UseConcMarkSweepGC -Xms1g -Xmx1g



* 1. -XX:+UseConcMarkSweepGC -Xms2g -Xmx2g



* 1. -XX:+UseConcMarkSweepGC -Xms4g -Xmx4g



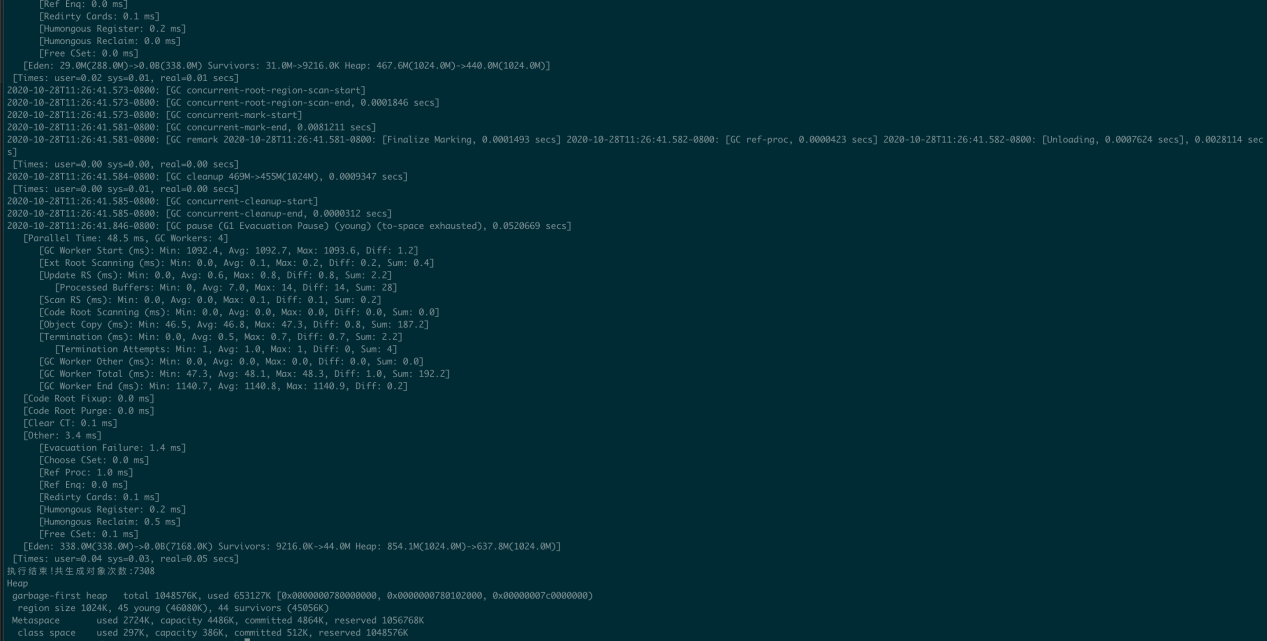
1. G1 GC
   1. -XX:+UseG1GC -Xms128m -Xmx128m



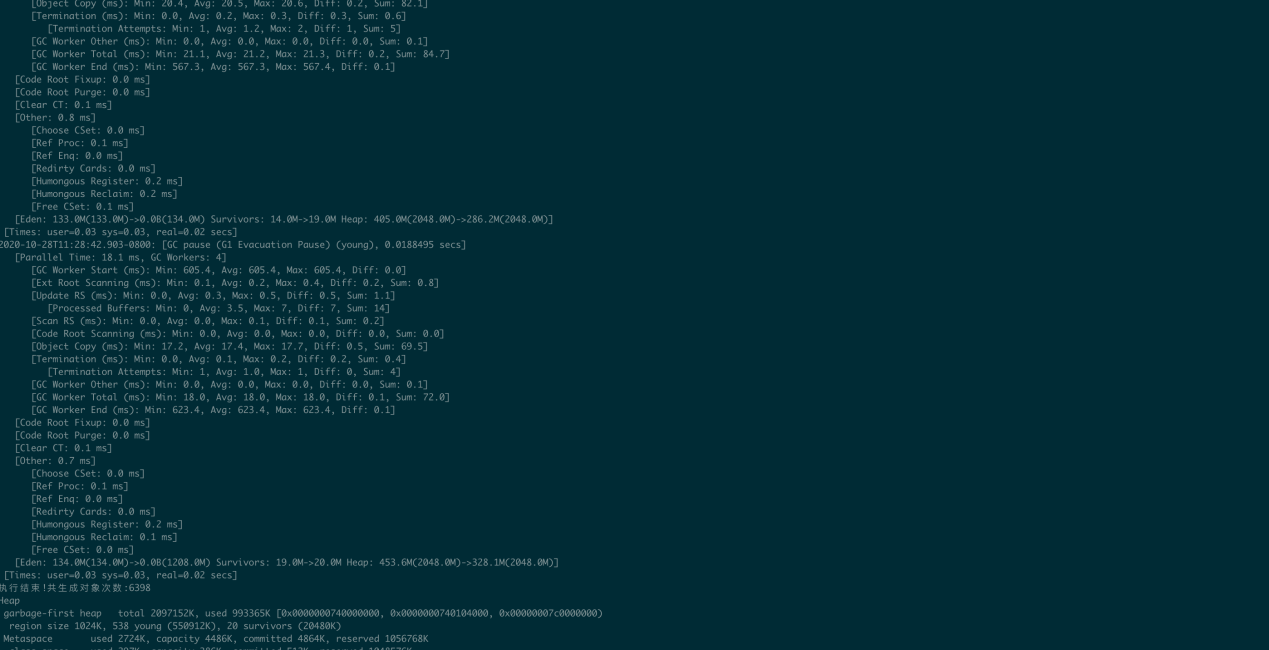
* 1. -XX:+UseG1GC -Xms512m -Xmx512m



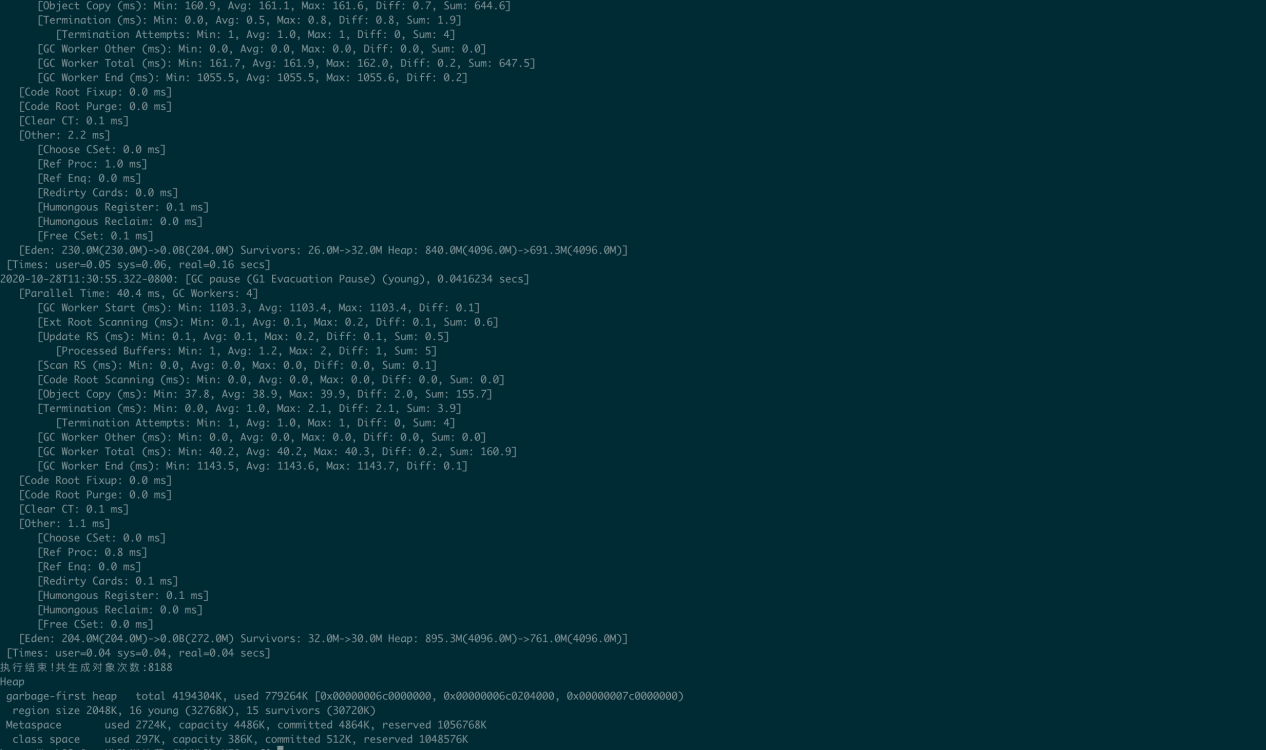
* 1. -XX:+UseG1GC -Xms1g -Xmx1g



* 1. -XX:+UseG1GC -Xms2g -Xmx2g



* 1. -XX:+UseG1GC -Xms4g -Xmx4g



1. 执行结果分析
2. 当Xmx设置为128m时，四种GC收集器均发生OOM，内存太小时不足以提供分配对象所需要内存时会发生OOM，并且OOM发生前会伴随多次GC，包含至少一次Full GC。
3. 当内存从128m逐步提升至1G过程中，GC发生的频率降低，甚至不会发生FullGC,但是每次GC的时间增长，这是由于堆内存本身够大的时候，有足够的内存分配对象所需空间，这样GC频次就会少，而每次发生GC时由于堆中已经生成了较多对象，相较于堆内存较小情况需要回收更多的空间，所以GC时间会稍长。
4. 在一次Full GC中，Young区的数据会大量被回收，相比之下Old区的数据回收比例较低，因为Old区中通常是一些需要长久存活的对象。
5. 内存较大情况下，CMS GC相较于并行GC，停顿时间更短，GC次数更多。