什么是装箱和拆箱：比如，

int i = 0；

object o；

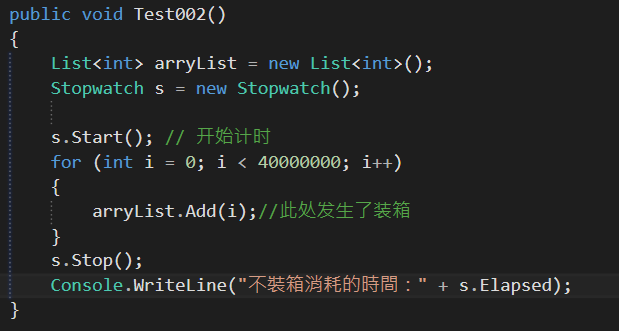
o=i; // 将具体的整型变量赋值给一个object,把子类的值赋值给了父类,这个过程就是装箱.

int j = (int)o;// 反之,就是拆箱.如果已经知道了object中存放的是int类型,那么拆箱就能够成功,否则可能会出现类型转换异常.

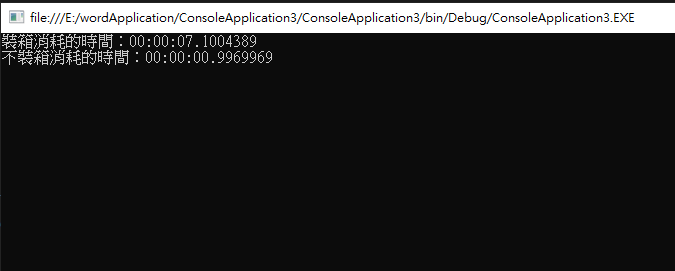
装箱和拆箱过程的性能问题:

将一个问题转换为另一个类型的过程中始终是有时间损耗的,所以不管是Java还是C#都引入了泛型的概念,来避免拆箱和装箱所带来的时间损耗问题.





显示结果：



其中的时间计时器StopWatch（）存放在System.Diagnostics命名空间中.

因此在数据量大,或者频繁操作的情况下最好还是避免拆箱和装箱操作,从时间上来看,装箱耗费的时间资源还是非常大的.