1.算法的效率：

算法的效率主要有以下两个复杂度来评估：

时间复杂度：评估执行程序所需的时间。可以估算出程序对处理器的使用程序

空间复杂度：评估执行程序所需的储存空间。可以估算出程序对内存的使用程度

2.时间复杂度：

时间频度：一个算法中的语句执行次数称之为语句频度或时间频度。记为T(n)

时间复杂度：算法中基本操作重复执行的次数是问题规模n的某个函数，用T(n)表示，若有某个辅助函数f(n)，使得当n趋近于无穷大时，T(n)/f(n)的极限值为不等于零的常数，则称f(n)时T(n)的同数量级函数，记作T(n)=O(f(n))，它称为算大的渐进时间复杂度，简称时间复杂度。

3.大O表示法

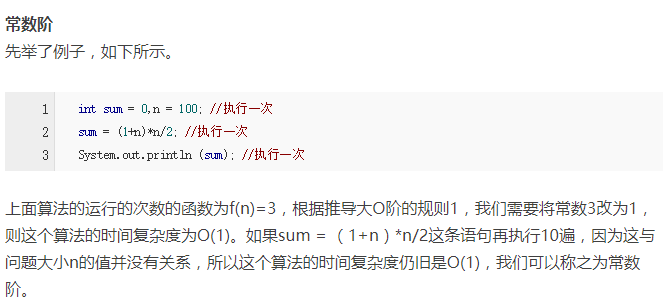
用O()来体现算法时间复杂度的记法称之为大O表示法

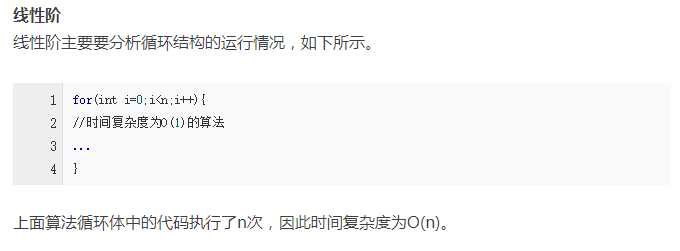
大O()表示发O(f(n))中的f(n)的值可以为1、n、logn、n2等，因此我们可以将O(1)、O(n)、O(log n)、O(n2)分别称为常数阶、线性阶、对数阶和平方阶。

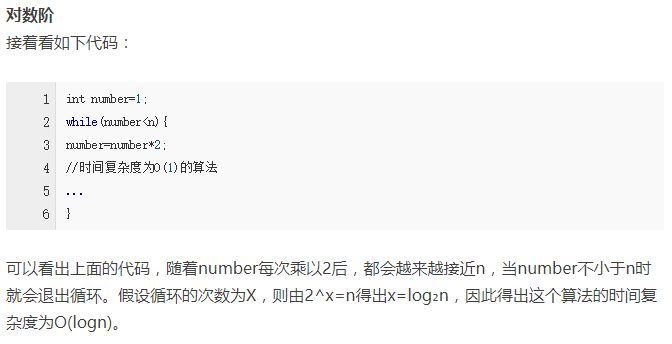
1. 推导大O阶

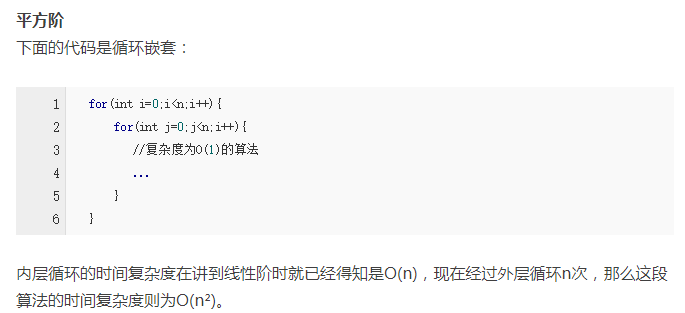
推导大O阶按照如下的规则来进行推导，得到的结果就是大O表示法：

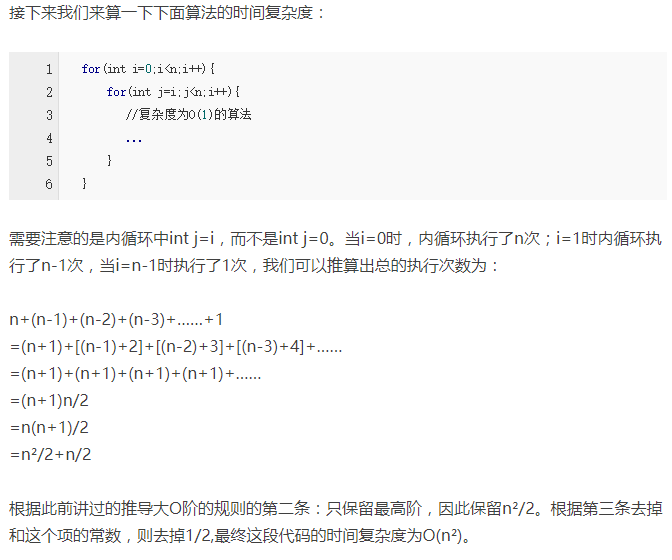
1. 用常数1来取代运行时间中所有的加法常数。
2. 修改后的运行次数函数中，只保留最高阶项
3. 如果最高阶项存在且不是1，则去除这个项相乘的常数











4．复杂度的比较

