判断算法性能，需要选用另一种尺度来代替运行时间。

可以考虑处理一定规模size时算法所需执行的基本操作basic operation数。

算法的增长率一般指输入值与算法代价的相关的单调递增函数T(n)。

在实时系统中，比较关注最差情况算法分析。在其他情况下，通常考虑平均情况.

估算一种算法运行的时间或者其他开销的增长率时，经常忽略其系数。

大O表示法：（同理下限可用大Ω表示法）

一般定义：如果某个算法的增长率（在最差情况下是f(n),那么就说这种算法“最差情況下在集合O(f(n))中“，直接说“最差情况下在O(n)中。

精确定义：对于非负函数T(n),如果存在两个正常数c和n0，对任意n>n0，有T(n)≤cf(n)，则称T(n)在集合O(f(n))中。

当上下限相等时可用Θ表示法，两个Θ相同的函数有交换性。

结构性开销( overhead)：一个数据结构的主要目的是用给当的方法存储数据，并且能够简单而有效地对其进行访问。为此，必须在这个数据结构中加上一些附加信息，指明数据存放在何处。

基于磁盘的空间/时间权衡原则是：在磁盘上的存储开销越小，程序运行得越快。这是因为从磁盘上读取数据的时间开销远远大于用于计算的时间开销。

**先调整算法，后调整代码！**