

1.2算法和算法评价

算法的基本概念

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中的每条指令表示一个或多个操作

重要特性

- 有穷性：一个算法必须执行有穷步之后结束，且每一步都可在有穷时间内完成
- 确定性：算法中每条指令必须有确切的含义，对于相同的输入只能得出相同的输出
- 可行性：算法中描述的操作都可以通过已经实现的基本运算执行有限次来实现
- 输入：一个算法有零个或多个输入，这些输入取自于某个特定的对象的集合
- 输出：一个算法有一个或多个输出，这些输出是与输入有着某种特定关系的量

优秀算法的标准

- 正确性：算法应能够正确地解决求解问题。
- 可读性：算法应具有良好的可读性，以帮助人们理解
- 健壮性：输入非法数据时，算法能适当地做出反应或进行处理，而不会产生莫名其妙的 输出结果
- 效率与低存储量需求：效率是指算法执行的时间，存储量需求是指算法执行过程中所需 要的最大存储空间，这两者都与问题的规模有关

算法效率的度量

算法效率的度量是通过时间复杂度和空间复杂度来描述的

时间复杂度

- 一个语句的频率是指该语句在算法中被重复执行的次数
- 常见的渐近时间复杂度为 $O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3) < O(2^n) < O(n!) < O(n^n)$

一般情况下，嵌套的循环次数是时间复杂度指数（存在特殊情况）

空间复杂度

- 算法的空间复杂度S(n)定义为该算法所耗费的存储空间,它是问题规模n的函数。
- 一个程序在执行时除需要存储空间来存放本身所用的指令、常数、变量和输入数据外，还需要一些对数据进行操作的工作单元和存储一些为实现计算所需信息的辅助空间
- 算法原地工作是指算法所需的辅助空间为常量，即O(1)

计算规则

- 加法规则： $O(f(n)) + O(g(n)) = O(\max(f(n), g(n)))$
- 乘法规则： $O(f(n)) \times O(g(n)) = O(f(n) \times g(n))$