



算法展望

【什么是算法】

算法 (Algorithm) 是解决特定问题的准确而完整的描述。算法是指令 (规则) 的有限序列。算法就是一种策略机制, 能够对一定规范的输入, 在有限时间内获得所要求的输出。

【时间复杂度】

算法的时间复杂度 (Time Complexity) 是指执行算法所需要的计算工作量, 一般来说是问题规模 n 的函数 $f(n)$ 。

算法的时间复杂度记做

$$T(n) = O(f(n))$$

常见的时间复杂度有: 常数阶 $O(1)$; 对数阶 $O(\log_2 n)$; 线性阶 $O(n)$; 线性对数阶 $O(n \log_2 n)$; 平方阶 $O(n^2)$; 立方阶 $O(n^3)$;; 指数阶 $O(2^n)$ 。

问题的规模 n 越大, 算法执行的时间也就越长。对于相同规模的问题, 时间复杂度越高, 算法的执行效率越低。

在计算时间复杂度的时候, 先找出算法的基本操作, 然后确定它的执行次数, 最高阶的执行次数就是时间复杂度。

【算法的基本结构】

一个算法可以用顺序、选择、循环三种基本结构组合而成。

【算法思想】

模拟; 枚举; 递推; 递归; 迭代; 贪心; 分治; 试探。

【数据结构】

向量; 栈; 队列; 链表; 树; 图; 堆 (优先队列); 森林; 集合; 映射。