

## Lesson3 (时空复杂度分析)

### 根据数据范围反推题目时间复杂度

一般ACM或者笔试题的时间限制是1秒或2秒。

在这种情况下，C++代码中的操作次数控制在  $10^7 \sim 10^8$  为最佳。

下面给出在不同数据范围下，代码的时间复杂度和算法该如何选择：

1.  $n \leq 30$ , 指数级别, dfs+剪枝, 状态压缩dp
2.  $n \leq 100 \Rightarrow O(n^3)$ , floyd, dp, 高斯消元
3.  $n \leq 1000 \Rightarrow O(n^2)$ ,  $O(n^2 \log n)$ , dp, 二分, 朴素版Dijkstra、朴素版Prim、Bellman-Ford
4.  $n \leq 10000 \Rightarrow O(n * \sqrt{n})$ , 块状链表、分块、莫队
5.  $n \leq 100000 \Rightarrow O(n \log n) \Rightarrow$  各种sort, 线段树、树状数组、set/map、heap、拓扑排序、dijkstra+heap、prim+heap、spfa、求凸包、求半平面交、二分、CDQ分治、整体二分
6.  $n \leq 1000000 \Rightarrow O(n)$ , 以及常数较小的  $O(n \log n)$  算法  $\Rightarrow$  单调队列、hash、双指针扫描、并查集、kmp、AC自动机, 常数比较小的  $O(n \log n)$  的做法: sort、树状数组、heap、dijkstra、spfa
7.  $n \leq 10000000 \Rightarrow O(n)$ , 双指针扫描、kmp、AC自动机、线性筛素数
8.  $n \leq 10^9 \Rightarrow O(\sqrt{n})$ , 判断质数
9.  $n \leq 10^{18} \Rightarrow O(\log n)$ , 最大公约数, 快速幂
10.  $n \leq 10^{1000} \Rightarrow O((\log n)^2)$ , 高精度加减乘除
11.  $n \leq 10^{100000} \Rightarrow O(\log k \times \log \log k)$ ,  $k$ 表示位数, 高精度加减、FFT/NTT

### 根据代码分析时间复杂度

如何求递归算法的时间复杂度（主定理）

如何求DFS与BFS的时间复杂度（每一层分别开）