

分治算法

一、设计思想（“分而治之”）

将一个难以直接解决的大问题，分割成一些规模较小的相同问题，以便各个击破，分而治之。

二、使用场景

1、特分治法所能解决的问题一般具有以下几个特征：

- 1) 该问题的规模缩小到一定的程度就可以容易地解决
- 2) 该问题可以分解为若干个规模较小的相同问题，即该问题具有最优子结构性质。（前提）
- 3) 利用该问题分解出的子问题的解可以合并为该问题的解；（关键）
- 4) 该问题所分解出的各个子问题是相互独立的，即子问题之间不包含公共的子子问题（效率）

第三条特征是关键，能否利用分治法完全取决于问题是否具有第三条特征，如果具备了第一条和第二条特征，而不具备第三条特征，则可以考虑用贪心法或动态规划法。

第四条特征涉及到分治法的效率，如果各子问题是不独立的则分治法要做许多不必要的工作，重复地解公共的子问题，此时虽然可用分治法，但一般用动态规划法较好。

三、分治法得基本步骤

step1 分解：将原问题分解为若干个规模较小，相互独立，与原问题形式相同的子问题；

step2 解决：若子问题规模较小而容易被解决则直接解，否则递归地解各个子问题

step3 合并：将各个子问题的解合并为原问题的解。

Divide-and-Conquer(P)

if $|P| \leq n_0$

then return(ADHOC(P))

将P分解为较小的子问题 P_1, P_2, \dots, P_k

for $i \leftarrow 1$ to k

do $y_i \leftarrow \text{Divide-and-Conquer}(P_i)$ \triangle 递归解决 P_i

$T \leftarrow \text{MERGE}(y_1, y_2, \dots, y_k)$ \triangle 合并子问题

return(T)

四、适用的经典问题

(1) 二分搜索

- (2) 大整数乘法
- (3) Strassen矩阵乘法
- (4) 棋盘覆盖
- (5) [合并排序](#)
- (6) [快速排序](#)
- (7) 线性时间选择
- (8) 最接近点对问题
- (9) 循环赛日程表
- (10) 汉诺塔