

# 递归于分治

- 1. 递归是直接或间接调用自身的一种方法
- 2. 递归出口，递推，回归
- 3. 递归与迭代的区别
  - 1. 递归会引起一系列的函数调用，同时还可能有大量的重复计算，效率较低
  - 2. 迭代是函数A不停地调用函数B的过程，与递归调用自己不一样
- 4. 分治的条件
  - 1. 当问题的规模较小是，可以较为容易的解决，
  - 2. 大问题可以分解为许多小规模，且**类型相同的子问题**，这是最优子结构性质
  - 3. 能够通过将子问题的解合并得到原问题的解
  - 4. 各个子问题之间是相互独立的，即子问题之间不存在公共子问题
- 5. 分治算法的时间复杂度的计算
  - 1. 迭代法，直接展开
  - 2. 迭代法-递归树法
  - 3. 主方法(渐进上届，下界，紧界)
- 6. 应用
  - 1. 二分查找
  - 2. 快排
  - 3. 二路归并排序

种类	时间复杂度	空间复杂度	稳定性	备注
快排	$n \log n$ $\sim n^2$	$O(n)$	不稳定	若序列本身有序或接近有序，快排退化为冒泡的 $O(n^2)$
二分查找	$\log n$			要求随机定位，有序
二路归并	$n \log n$	$O(n)$	稳定	

## 7. 大整数乘法