1. **分治法**

分治法的设计思想是，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。

**1、什么是分治算法**

分治算法是：（分而治之）

* 原问题成互不相交的子问题
* 同样的方法求解子问题
* 把子问题的解合并成原问题的解

**2、分治法所能解决的问题一般具有以下几个特征：**

* 该问题的规模缩小到一定的程度就可以容易地解决；
* 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。
* 利用该问题分解出的子问题的解可以合并为该问题的解；
* 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。

上述的第一条特征是绝大多数问题都可以满足的，因为问题的计算复杂性一般是随着问题规模的增加而增加；第二条特征是应用分治法的前提，它也是大多数问题可以满足的，此特征反映了递归思想的应用；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。

**例子**：棋盘覆盖、大整数相乘、矩阵相乘

**二、动态规划**

**1、什么是动态规划**

动态规划：与分治法相似，把问题分解按层次分成子问题，直到可以直接求解的子问题，然后一级一级地向上求解。

与分治法的出别在于：动态规划适用有许多重复子问题出现的问题，它保留已求出问题的解。

**2、运用动态规划需符合的条件**

1）、最优化原理

最优化原理：一个最优化策略具有这样的性质，不论过去状态和决策如何，对前面的决策所形成的状态而言，余下的诸决策必须构成最优策略。简而言之，一个最优化策略的子策略总是最优的。

2）、无后效性

“过去的步骤只能通过当前状态影响未来的发展，当前的状态是历史的总结”。这条特征说明动态规划只适用于解决当前决策与过去状态无关的问题。状态，出现在策略任何一个位置，它的地位相同，都可实施同样策略，这就是无后效性的内涵。

由上可知，最优化原理，无后效性，是动态规划必须符合的两个条件。

**3、动态规划的基本思路**

对于一道题，怎样具体运用动态规划方法呢？

1. 首先，分析题意，考察此题是否满足最优化原理与无后效性两个条件。
2. 接着，确定题中的阶段，状态，及约束条件。
3. 推导出各阶段状态间的函数基本方程，进行计算。

具体求解则有多种方法。

* 1. 前向与后向动态规划法：所谓前向与后向，指的是从起点出发，层层递推，直到终点，或从终点出发，逆向求解。这两种方法本质上是一样的，具体解题时，可根据实际情况来选用。
  2. 具有隐含阶段的问题（即阶段不明显）：动态规划的一个重要环节是阶段划分，可有些题目无明显阶段，但也符合最优化原理。

**三 贪心算法**

贪心算法：总是做出在当前来看是最好的选择。就是说，贪心并不从整体最优上来考虑，所做出的选择只是某种意义上的局部最优的选择。当然希望贪心算法得到的最终结果是最优的。可是贪心算法并不能保证最终结果是最优的。不过，在许多的情况下，应用贪心算法能够得到整体最优解；并且在一些情况下，即使得到的不是最优解，也是一个很好的近似解

**1．贪心法的基本思路：**

从问题的某一个初始解出发逐步逼近给定的目标，以尽可能快的求得更好的解。当达到某算法中的某一步不能再继续前进时，算法停止。(不能回溯)

该算法存在问题：

1)、不能保证求得最后的解是最佳的；

2)、不能用来求最大或最小解问题；

3)、只能求出满足某些约束条件的可行解的范围。

**2. 二个基本要素**

1)、贪心选择性质：

选择具有无后效性，即不依赖与以后将要作出的选择。

问题的整体最优解可通过一系列的局部最优选择来达到。

证明方法：用归纳法证明此问题的最优解包括一个贪心选择的解。

2)、最优子结构性质

一个问题的最优解包括其子问题的最优解。

贪心算法(greedy method)就是:只顾眼前利益，每次都选最好的，是由数学家菲波那契提出，有的教材又叫贪婪算法。

**四、分支限界法：**

1. **什么是分支限界法**

二个基本概念：活结点、扩展结点。

分支限界法类似于回溯法，也是一种在问题的解空间树T上搜索问题解的算法。但在一般情况下，分支限界法与回溯法的求解目标不同。回溯法的求解目标是找出T中满足约束条件的所有解，而分支限界法的求解目标则是找出满足约束条件的一个解(八皇后问题，92种解全部求出)。

由于求解目标不同，导致分支限界法与回溯法在解空间树T上的搜索方式也不相同。回溯法以深度优先的方式搜索解空间树T，而分支限界法则以广度优先的方式搜索解空间树T。

分支限界法的搜索策略是：在扩展结点处，先生成其所有的儿子结点（分支），然后再从当前的活结点表中选择下一个扩展结点。为了有效地选择下一扩展结点，以加速搜索的进程，在每一活结点处，计算一个函数值（限界），并根据这些已计算出的函数值，从当前活结点表中选择一个最有利的结点作为扩展结点，使搜索朝着解空间树上有最优解的分支推进，以便尽快地找出一个最优解。

**2、分支限界法的基本思想**

分支限界法常以广度优先的方式搜索问题的解空间树。问题的解空间树是表示问题解空间的一棵有序树。在搜索问题的解空间树时，分支限界法与回溯法对当前扩展结点所使用的扩展方式不同。在分支限界法中，每一个活结点只有一次机会成为扩展结点。活结点一旦成为扩展结点，就一次性产生其所有儿子结点。在这些儿子结点中，那些导致不可行解的儿子结点被舍弃，其余儿子结点被子加入活结点表中。此后，从活结点表中取下一结点成为当前扩展结点，并重复上述结点扩展过程。这个过程一直持续到找到所求的解或活结点表为空时为止。

选择下一个扩展结点的方法：

先进先出（FIFO），即将活结点组织成一个队列，并按队列的先进先出的原则选取一个结点为当前扩展结点。

**3、分支限界法的实现：**

分支限界法就是最佳优先(包括广度优先在内)的搜索法。

**五 回溯法**

**1.什么是回溯法**

回溯法实际是穷举算法，按问题某种变化趋势穷举下去，如某状态的变化用完还没有得到最优解，则返回上一种状态继续穷举。回溯法有“通用的解题法”之称，其采用了一种“走不通就掉头”思想作为其控制结构，用它可以求出问题的所有解和任意解。

它的应用很广泛，很多算法都用到回溯法，例如，迷宫，八皇后问题，图的m着色总是等都用到回溯法，当然其中还使用了其他策略。（教材中列出了解18种问题）

**2.基本思想**

回溯法从开始结点（根结点）出发，以深度优先的方式搜索整个解空间（一般为树结构空间）。这个开始结点就成为一个活结点，同时也成为当前的扩展结点。在当前的扩展结点处，搜索向纵深方向移至一个新结点。这个新结点就成为一个新的活结点，并成为当前扩展结点。如果在当前的扩展结点处不能再向纵深方向移动，则当前扩展结点就成为死结点。换句话说，这个结点不再是一个活结点。此时，应往回移动（回溯）至最近的一个活结点处，并使这个活结点成为当前的扩展结点。回溯法即以这种工作方式递归地在解空间中搜索，直至找到所要求的解或解空间中已没有活结点时为止。

运用回溯法解题通常包含以下三个步骤：

（1）针对所给问题，定义问题的解空间；

（2）确定易于搜索的解空间结构；

（3）以深度优先的方式搜索解空间，并且在搜索过程中用剪枝函数避免无效搜索；

**贪心算法与动态规划算法**

多阶段逐步解决问题的策略就是按一定顺序或一定的策略逐步解决问题的方法。分解的算法策略也是多阶段逐步解决问题策略的一种表现形式，主要是通过对问题逐步分解，然后又逐步合并解决问题的。

  贪心算法每一步都根据策略得到一个结果，并传递到下一步，自顶向下，一步一步地做出贪心决策。

  动态规划算法的每一步决策给出的不是唯一结果，而是一组中间结果，而且这些结果在以后各步可能得到多次引用，只是每走一步使问题的规模逐步缩小，最终得到问题的一个结果。