



USACO 习题总结

前 言

USACO 全称美国计算机奥林匹克竞赛，其官方网站所开设的训练系统 USACO Training，是全球知名的信息学在线题库。该题库拥有很高的题目质量，且难度由浅及深层次清晰，使得不同水平的竞赛选手均能从中获益。本文即是笔者历时数月完成 USACO Training 中全部题目后所做的习题总结，以供日后复习和参考之用。

USACO 题库经过多年的维护与修订，现共有 6 章 23 节 97 题，其中多含历届国际大赛真题，所涉及的算法和数据结构之广，也几乎完整涵盖了国内分区联赛的所有题型。故而民间向来就有“做完前三章，稳拿一等奖”的说法，足见该题库在竞赛学习中的价值。然而更为重要的是：许多学校在开展竞赛教学的过程中，往往过度偏重于知识的讲解与传授，却忽视了习题的重要性，选手的模型还原能力和算法实现能力无法得到真正的锻炼，则势必会严重影响其实力的发挥；但若一味地埋头题海，对高级算法和数据结构缺乏应有的了解，也难以取得理想的竞赛成绩。所以循序渐进，在掌握新算法的同时适时巩固练习，未尝不是最为科学合理学习方法，而 USACO 题库则正是这样一套优秀的竞赛训练系统。

相信无论是信息竞赛的初学者，还是冲刺奖牌的优胜者，抑或是程序设计的爱好者，都能在完成这样一套题库之后收获和巩固自己对算法的认知与感悟。虽然我个人水平相当有限，但假如这套习题总结，能为您在竞赛学习的过程中提供些许帮助，我必将深感荣幸。

重庆南开中学 万山川

2009 年 12 月

目 录

一、题目索引	P03
二、简明题解	P08
1) Chapter1	08
2) Chapter2	12
3) Chapter3	16
4) Chapter4	21
5) Chapter5	25
6) Chapter6	30
附：参考资料	P31

一、题目索引

第一章

Chapter1 Getting started

1.1.1	Your Ride Is Here	字符串处理
1.1.2	Greedy Gift Givers	模拟
1.1.3	Friday the Thirteenth	模拟
1.1.4	Broken Necklace	枚举
1.2.1	Milking Cows	模拟
1.2.2	Transformations	模拟
1.2.3	Name That Number	模拟
1.2.4	Palindromic Squares	进制转换
1.2.5	Dual Palindromes	进制转换
1.3.1	Mixing Milk	贪心
1.3.2	Barn Repair	贪心
1.3.3	Calf Flac	枚举
1.3.4	Prime Cryptarithm	枚举
1.4.1	Packing Rectangles	模拟
1.4.2	The Clocks	搜索
1.4.3	Arithmetic Progressions	枚举
1.4.4	Mother's Milk	搜索
1.5.1	Number Triangles	动态规划
1.5.2	Prime Palindromes	筛法
1.5.3	SuperPrime Rib	枚举
1.5.4	Checker Challenge	搜索

第一章是为初学者们所开设的章节，只会涉及到一些基础算法，诸如模拟、枚举、贪心和简单的动态规划。适合初学者们实践从中课本中学到的语法知识，使选手们更进一步了解到竞赛题目的风格。

第二章

Chapter2 Bigger Challenges

2.1.1	The Castle	灌水法
2.1.2	Ordered Fractions	枚举
2.1.3	Sorting A Three-Valued Sequence	贪心
2.1.4	Healthy Holsteins	搜索
2.1.5	Hamming Codes	位运算
2.2.1	Preface Numbering	枚举
2.2.2	Subset Sums	动态规划
2.2.3	Runaround Numbers	模拟
2.2.4	Party Lamps	搜索
2.3.1	The Longest Prefix	动态规划
2.3.2	Cow Pedigrees	动态规划
2.3.3	Zero Sum	搜索
2.3.4	Money Systems	动态规划
2.3.5	Controlling Companies	搜索
2.4.1	The Tamworth Two	模拟
2.4.2	Overfencing	搜索
2.4.3	Cow Tours	最短路径
2.4.4	Bessie Come Home	最短路径
2.4.5	Fractions to Decimals	模拟

第二章加入了更多对编程技巧的考验，有些题则会用到与之前相比更为高级的算法，最为实用的当属最短路径算法和位运算优化。本章中动态规划和搜索也是考察的重点，尤其体现在第三小节，而这也正是国内联赛中最热门的题型，应当熟练掌握。

第三章

Chapter3 Techniques more subtle

3.1.1	Agri-Net	最小生成树
3.1.2	Score Inflation	动态规划
3.1.3	Humble Numbers	枚举
3.1.4	Shaping Regions	矩形切割
3.1.5	Contact	字典树
3.1.6	Stamps	动态规划
3.2.1	Factorials	高精度
3.2.2	Stringsobits	动态规划
3.2.3	Spinning Wheels	模拟
3.2.4	Feed Ratios	枚举
3.2.5	Magic Squares	搜索
3.2.6	Sweet Butter	最短路径
3.3.1	Riding The Fences	欧拉路径
3.3.2	Shopping Offers	动态规划
3.3.3	Camelot	最短路径
3.3.4	Home on the Range	动态规划
3.3.5	A Game	博弈动规
3.4.1	Closed Fences	计算几何
3.4.2	American Heritage	递归
3.4.3	Electric Fence	枚举
3.4.4	Raucous Rockers	动态规划

第三章中会接触到越来越多的新题型和新算法，诸如欧拉路、最小生成树、矩形切割、线段树以及计算几何中的一些基本公式。很多题目并不局限于唯一的标准解法，选手在习题过程中应当勇于尝试以不同或更优的算法解决相同的问题。

第四章

Chapter4 Advanced algorithms and difficult drills

4.1.1	Beef McNuggets	动态规划
4.1.2	Fence Rails	迭代加深
4.1.3	Fence Loops	最小环
4.1.4	Cryptcowgraphy	搜索
4.2.1	Drainage Ditches	最大费用流
4.2.2	The Perfect Stall	二分图匹配
4.2.3	Job Processing	贪心
4.2.4	Cowcycles	搜索
4.3.1	Buy Low, Buy Lower	动态规划
4.3.2	The Primes	搜索
4.3.3	Street Race	搜索
4.3.4	Letter Game	枚举
4.4.1	Shuttle Puzzle	构造法
4.4.2	Pollutant Control	最小边割
4.4.3	Frame Up	拓扑排序

第四章题目的整体难度明显加强，网络流、二分图等高级算法和数据结构开始登场，而搜索的优化和剪枝也开始变得越发重要。选手们很难再找到一道不需要高级算法或进一步优化便能轻松解决的题目，这为我们的竞赛学习带来了新的挑战 and 更高的要求。

第五章、第六章

Chapter5 Serious challenges

5.1.1	Fencing the Cows	二维凸包
5.1.2	Starry Night	搜索
5.1.3	Musical Themes	动态规划
5.2.1	Snail Trail	搜索
5.2.2	Electric Fences	搜索
5.2.3	Wisconsin Squares	搜索
5.3.1	Milk Measuring	迭代加深
5.3.2	Window Area	矩形切割
5.3.3	Network of Schools	强连通分量
5.3.4	Big Barn	动态规划
5.4.1	All Latin Squares	搜索
5.4.2	Canada Tour	动态规划
5.4.3	Character Recognition	动态规划
5.4.4	Betsy's Tour	搜索
5.4.5	TeleCowmunication	最小点割
5.5.1	Picture	线段树
5.5.2	Hidden Passwords	枚举
5.5.3	Two Five	记忆化搜索

Chapter6 Contest Practice

6.1.1	Postal Vans	动态规划
6.1.2	A Rectangular Barn	动态规划
6.1.3	Cow XOR	位运算

第五章、第六章拥有与第四章相似的特点和更高的难度，每一道题都有很高的训练价值，有的甚至还饶有趣味，非常适合联赛级别以上的选手逐一攻克。

二、简明题解

Chapter1 Getting started

1.1.1 Your Ride Is Here (ride)

题目 计算并比较两字符串每位字符 ASCII 值的连续乘积。

题型 字符串处理

题解 直接模拟即可。

1.1.2 Greedy Gift Givers (gift1)

题目 已知每人的钱数和好友，将钱均分给好友后，求每人所得结余。

题型 模拟、字符串哈希*

题解 直接模拟即可，检索姓名时可以使用字符串哈希优化。

1.1.3 Friday the Thirteenth (friday)

题目 统计一定年份内每月 13 日在星期一至日出现的次数。

题型 模拟、蔡勒公式*

题解 以月为最小单位模拟计算即可，也可用蔡勒公式根据日期直接推算出星期。

1.1.4 Broken Necklace (beads)

题目 计算一串由三种不同字符组成的环中，从任意位置起，向左右两旁可拓展的最长距离和。

题型 枚举、动态规划*

题解 枚举起始位置并计算扩展距离。

1.2.1 Milking Cows (milk2)

题目 给出 N 个区间，求所有区间并集的补集。

题型 模拟、离散化*、线段树*

题解 用布尔数组模拟即可，但需注意所给数据均为开区间，可将数组范围展开至 2 倍，将每个点拆为起止两个点。

1.2.2 Transformations (transform)

题目 给出两幅字符串组成的图形和 7 种变换规则，求两图形间变换方法。

题型 模拟

题解 枚举 7 种变换规则，寻找匹配方案。

1.2.3 Name That Number (namenum)

题目 给出一串数字、一份字符串表和数字与字母之间的对应规则，求表中哪些字符串符合对应规则。

题型 模拟

题解 因字符串表保证升序排列，可直接一边读入一边判断。

1.2.4 Palindrome Squares (palsquare)

题目 输出 B 进制下 300 以内所有 B 进制二次方结果成回文状的数。

题型 进制转换

题解 模拟转换过程即可。

1.2.5 Dual Palindromes (dualpal)

题目 输出前 N 个大于 S 且在两种及以上进制中均成回文状的数。

题型 进制转换

题解 因数据量允许，可直接枚举 S 以上 10 进制数。

1.3.1 Dual Palindromes (dualpal)

题目 已知同种物品在不同商家处的单价和供应量，求满足额定需求量所支出的最小花费。

题型 贪心

题解 尽可能从价廉的厂商处优先买进，直至满足需求。

1.3.2 Barn Repair (barn1)

题目 数轴上有 C 个被标记点，求用 M 个区间覆盖住所有被标记点所需的最小区间长度。

题型 贪心、动态规划*

题解 最小覆盖长度等同于最大未覆盖长度，初始时可令一大区间覆盖数

轴上全部点集，按次截去现有区间中最长未标记子区间，将木板截为 M 段即可。

1.3.3 Calf Flac (calfflac)

题目 寻找最长回文子串。

题型 枚举、动态规划*

题解 枚举子串起始位置，并验证回文属性。

1.3.4 Prime Cryptarithm (crypt1)

题目 以给定数字填充固定格式的竖式乘法表，求合法的方案数。

题型 枚举、搜索

题解 枚举被乘数和乘数共计 5 个数位上的数，并验证方案合法性。

1.4.1 Packing Rectangles (packrec)

题目 给定 4 个矩形块，找出一个最小的封闭矩形将这 4 个矩形块放入，但不得相互重叠。

题型 模拟

题解 模拟 6 种可能的摆放方案，求出最小边长。

1.4.2 The Clocks (clocks)

题目 有 9 块时针指向不同的时钟，以及 9 种不同的调整操作，求将所有时钟时针调至 12 点方向的最短操作顺序。

题型 搜索、枚举

题解 经典的宽度优先搜索例题，需使用哈希判重，对于此题深搜与暴力枚举也同样适用。

1.4.3 Arithmetic Progressions (ariprog)

题目 在双平方数集合（即所有能表示成 $p^2 + q^2$ 的数的集合）中长度为 N 的等差数列。

题型 枚举

题解 枚举所有可能解，适时剪枝即可。

1.4.4 Mother's Milk (milk3)

题目 ABC 三个杯子相互倒灌，初始时 AB 为空 C 为满，求 C 杯子任意时刻可能的剩余量。

题型 搜索

题解 宽搜判重，枚举每个时刻可能的 6 种倒灌方案。

1.5.1 Number Triangles (numtri)

题目 现有一个数字金字塔，求一条从最高点到底部任意处结束的路径，使得所经过的数字之和最大。

题型 动态规划

题解 经典的动态规划，令 $f[i,j]$ 为到达第 j 层第 i 个位置时所取得的最大值，转移方程为 $f[i,j]=\max\{f[i,j-1],f[i-1,j-1]\}+a[i,j]$ 。

1.5.2 Prime Palindromes (pprime)

题目 找出一定范围内的所有回文素数。

题型 枚举

题解 先用枚举求出范围内全部回文数，再判断是否为素数。

1.5.3 SuperPrime Rib (sprime)

题目 求出所有 N 位特殊质数，即该数前 $1\sim N$ 位均为质数。

题型 搜索

题解 深搜构造每一位并判断是否为质数，首位只能由 2,3,5,7 构成，其余为只可能为 1,3,7,9。

1.5.4 Checker Challenge (checker)

题目 N 皇后问题，输出前三个解的方案和解的总数。

题型 搜索、位运算优化

题解 使用位运算优化的搜索题。在搜索过程中分别用三个二进制参数记录竖列和两条对角线的摆放情况，将已放置的位置标记为 $1_{(2)}$ 。可由这三个参数的并集的补集得到所在横行剩余的合法位置，每次取右边第一个合法位置递归深入，当参数中所有二进制数位均被标记为 $1_{(2)}$ 时即找到了一种合法方案。

Chapter2 Bigger Challenges

2.1.1 The Castle (castle)

题目 平面房间被墙分成多个小区域，求最大的两块相邻的连通区域。

题型 灌水法、位运算、并查集*

题解 输入数据中的数字完全按照二进制规则给出，从高位向低位依次表示西北东南四个方向上是否有围墙，用灌水法搜出每块连通区域面积，再枚举相邻区域使面积之和最大即可。

2.1.2 Ordered Fractions (frac1)

题目 找出所有满足条件的最简分数 a/b ，使得 $1 \leq b \leq N$ 且 $0 \leq a/b < 1$ 。

题型 枚举、排序、递归*

题解 枚举所有符合条件的 a 和 b ，并按照 a/b 排序输出。

2.1.3 Sorting A Three-Valued Sequence (sort3)

题目 一个由 1,2,3 组成的数字序列，求排成升序所需的最少交换次数。

题型 贪心

题解 升序序列必按出现次数分三段排列，每次可将第一段中非 1 数与原所属段或另一段中的 1 相交换，并依此类推交换所有乱序数字。

2.1.4 Healthy Holsteins (holstein)

题目 每件物品有 V 种附加值，求最少的选择方案，使得 V 种附加值的需求量均得到满足。

题型 搜索、迭代加深*

题解 深搜每样物品是否被选择，逐次加大搜索深度，并记录最优方案。

2.1.5 Hamming Codes (hamming)

题目 从所有非负数中找出 N 个编码，使得编码两两之间在二进制表示法中至少有 D 个数位上的数字不相同。

题型 枚举、位运算

题解 从 0 开始枚举，使得每个新加入输出列表的数，与之前找出的所有数比对均符合条件。

2.2.1 Preface Numbering (preface)

题目 统计在用罗马数字表示的前 N 个自然数中 IVXLCDM 的出现次数。
题型 枚举、数学方法*
题解 枚举所有 N 以内的数，将其转为罗马数字后统计字母的出现次数。

2.2.2 Subset Sums (subset)

题目 对于前 N 个自然数组成的集合，可将其划分为两个不相交的子集合，且保证两个集合内的数字和相等，求划分方案的总数。
题型 动态规划
题解 等同于求物品大小为 $1 \sim N$ 而总容量为 $(n+1)*n/4$ 时的背包问题的方案数，可令 $f[j]$ 表示子集之和为 j 时的方案数，逐一枚举前 N 个自然数 i 和背包容量 j ， $f[j] = \sum (f[j-i])$ 。

2.2.3 Runaround Numbers (runround)

题目 求第一个大于 M 且具有这样性质的循环数：从第 2 个数开始循环地数第一个数的大小个数，以后每次从停止位后开始数停止位上数的长度的数，在经过每个数字一次后回到起点的就是循环数。
题型 模拟
题解 从 M 开始枚举，找到第一个循环数即可。

2.2.4 Party Lamps (lamps)

题目 有 N 盏灯和 4 种批量开关灯的操作方式，给出经过 C 次以内的操作之后部分灯的状态，求所有灯最后可能的状态。
题型 搜索、位运算*
题解 搜索即可，可以用位运算优化，根据规律只需用前 4 盏灯即可表示所有灯的状态，并且使用 xor 异或可以很方便地模拟 4 种开关。

2.3.1 The Longest Prefix (prefix)

题目 给出一个字符串 S 和一个由字串组成的集合，求 S 中可由集合内元素所构成的最长前缀长度。
题型 动态规划、字典树*
题解 $f[i]$ 表示长度为 i 的前缀能否取得， $len[j]$ 表示集合内字串长度， $f[i] = f[i-len[j]]$ and $(copy(s, i-len[j]+1, len[j]) = a[j])$ 。

2.3.2 Cow Pedigrees (nocows)

题目 求有 N 个结点, 深度为 M 且每个结点的度均为偶数的二叉树个数。

题型 动态规划

题解 令 $f[i,j]$ 表示结点数为 i 深度为 j 的二叉树个数, 由乘法原理可得,
 $f[i,j] = \sum (f[k,j-1] * f[i-k-1,j-1]) \quad (1 \leq k \leq i-2)$ 。

2.3.3 Zero Sum (zerosum)

题目 在 $1 \sim N$ 的序列中插入加减运算符或者空格, 使得算式结果为 0。

题型 搜索

题解 深搜枚举运算符, 并验证结果是否为零。

2.3.4 Money Systems (money)

题目 给出不同货币的面值, 求构成一定数额的方案数。

题型 动态规划

题解 令 $f[i]$ 表示钱数为 i 时的方案数, $f[i] = \sum (f[i-a[j]])$ 。

2.3.5 Controlling Companies (concom)

题目 当公司 A 拥有 B 一半以上的股权或 A 的子公司在 B 的股权之和超过 50% 时 A 可以控制 B , 给出股权分布情况, 求公司间控制情况。

题型 搜索

题解 根据规则循环更新被控制情况, 直至更新停止。

2.4.1 The Tamworth Two (ttwo)

题目 F 和 C 均按固定的前进方式在地图里游荡, 求最短相遇时间。

题型 模拟

题解 模拟两者前行方式, 并用所在位置判重即可。

2.4.2 Overfencing (maze1)

题目 一个有多个出口的迷宫, 求从任意一点走出迷宫的最小步数。

题型 搜索

题解 用 $dis[i,j]$ 记录从 (i,j) 走出迷宫所需最小步数, 依次从每个入口开始宽搜, 不断更新 $dis[i,j]$, 答案即是 $dis[i,j]$ 中的最大值。

2.4.3 Cow Tours (cowntour)

题目 给出一些独立的无向图，并定义图的直径为任意两点之间最短路径的最大值，现在任意两图中连接一条边，求连通后新图的最小直径。

题型 枚举、多源最短路

题解 先求出每个图内的最短路径，并用 $\text{dist}[i]$ 记录点 i 到相距最远的点的最短距离，再枚举分属两个不同图内的点 i 和 j ，新图的最小直径即 $\min(\text{dist}[i] + \text{dist}[j] + \text{dis}(i, j))$ ，其中 $\text{dis}(i, j)$ 为两点间距离。

2.4.4 Bessie Come Home (comehome)

题目 求图中一点到其他被标记点最短路径的最小值。

题型 单源最短路

题解 经典的最短路模型，可尝试用尽量多的最短路算法解决此题，并且因数据范围较小，甚至 $O(n^3)$ 的 Floyd 算法对此题也同样适用。

2.4.5 Fractions to Decimals (fracdec)

题目 将给出的分数转换为小数形式，可能会出现循环小数。

题型 模拟

题解 模拟长除法过程，当余数不为 0 时，便将其扩大 10 倍作为新的被除数，重复此过程，直到被除尽或找到循环节。记录下每个余数第一次出现的位置，若相同的余数重复出现，则当前位置与初次出现位置之间的部分即是循环节。

*Chapter3 Techniques more subtle***3.1.1 Agri-Net (agrinet)**

题目 用 $N-1$ 条边将 N 个点连接起来, 求边长和最短的方案。

题型 最小生成树

题解 经典的最小生成树例题, 常用算法有适用于稠密图的 Prim 和适用于稀疏图 Kruskal 两种。

3.1.2 Score Inflation (inflate)

题目 在总容量一定的情况下, 求物品选择方案使得总价值最大, 每种物品可重复选择。

题型 动态规划

题解 完全背包问题, $f[i]$ 表示总容量为 i 时总价值的最大值, 状态转移方程与 0/1 背包相同, 只是枚举容量时应从下限向上枚举, $f[j]=\max(f[j-\text{vol}[i]]+\text{val}[i])$ 。

3.1.3 Humble Numbers (humble)

题目 现有素数集合 $S1$, 求合数集合 $S2$, 其中所有合数的质因数均为 $S1$ 中素数, 并且这样的合数被称为丑数, 求 $S2$ 中第 N 个丑数。

题型 枚举

题解 枚举素数与丑数集合间的最小乘积, 逐个产生新的丑数。对于每个素数, 可以分别记录上一次与第几个丑数相乘, 避免重复求积。

3.1.4 Shaping Regions (rect1)

题目 将 N 个不同颜色的矩形相互重叠地放置在白色平面上, 求各种颜色最后剩余的可见面积。

题型 矩形切割、线段树*

题解 本题坐标范围极大, 普通的灌水法和坐标离散化都会超过空间或时间限制, 可以考虑采用矩形切割算法解决, 将新加入的矩形与队中其他矩形相交的部分切割后放入队中, 并更新重叠部分的颜色标记, 最后统计队中各矩形面积即可。

3.1.5 Contact (contact)

- 题目** 给出一串 01 字符，求其中长度在 A 到 B 之间的子串的重复频率。
- 题型** 字典树、散列表
- 题解** 由于字符串中只有 0 和 1 并且 $B \leq 12$ ，可以采用字典树很方便地解决本题，对于每个的长度在 A~B 之间的子串，对其在树中所在链上叶子结点的计数累加 1，最后统计所有叶子上的计数结果即可。

3.1.6 Stamps (stamps)

- 题目** 已知一个 N 枚邮票的票面集合和张数使用上限 K，求此集合可组成的从 1 开始连续的邮资集合。
- 题型** 动态规划
- 题解** 令 $f[i]$ 表示组成价值为 i 的邮资所需的最少邮票张数，易得出状态转移方程 $f[i] = \min(f[i - \text{val}[j]] + 1, f[i])$ ，当某一时刻 $f[i]$ 的值大于上限 K 时即求得答案为 $i - 1$ 。

3.2.1 Factorials (fact4)

- 题目** 求 $N!$ 最后一位非零数， $N \leq 4220$ 。
- 题型** 高精度
- 题解** 因数据范围较小，直接用高精度乘法求该数阶乘即可。

3.2.2 Stringsobits (kimbits)

- 题目** 求第 l 个二进制长度为 N 且 $1_{(2)}$ 的个数不超过 L 的数。
- 题型** 动态规划、康托展开
- 题解** 令 $f[j, k]$ 表示二进制长度为 j 且 1 的个数不超过 k 个的数的总数， $f[j, k] = f[j - 1, k] + f[j - 1, k - 1]$ ，即在当前位加上 0 和 1 两种状态，最后再用康托展开求出第 l 个数的二进制表示。

3.2.3 Spinning Wheels (spin)

- 题目** 五个不同转速的轮子，每个轮子上都有一处缺口，求五处缺口初次重合所需时间。
- 题型** 模拟
- 题解** 模拟轮轴转动过程即可。

3.2.4 Feed Ratios (ratios)

题目 市场上每份饲料由三种原料所构成, 现用三种原料配比不同的饲料调配理想饲料, 求方案。

题型 枚举

题解 枚举每种饲料的份数, 记录符合理想配比的最小方案。

3.2.5 Magic Squares (msquare)

题目 给出 2×4 个有序方格和三种基本操作, 求转换到目标状态所需的最少操作序列。

题型 搜索、康托展开

题解 根据三种操作宽度优先搜索, 用康托展开判重。

3.2.6 Sweet Butter (butter)

题目 求图中一点, 使得该点到其他点的最短路径和最短。

题型 最短路径

题解 枚举集结点, 求单源最短路长度和。

3.3.1 Riding The Fences (fence)

题目 从任意点开始遍历图中所有边, 求字典序最小的途径顶点序列。

题型 欧拉路径

题解 遍历边的过程即是求一条欧拉路径的过程对于无向图, 如果每个点的度都是偶数, 那么图中存在一条欧拉回路, 遍历的起点可以是任意点; 如果有且仅有 2 个点的度为奇数, 那么图中存在一条欧拉路, 遍历的起点只能是度为奇数的点。

3.3.2 Shopping Offers (shopping)

题目 商店将部分物品捆绑销售并附有价格优惠, 现给出每件物品的需求量, 求刚好满足需求量的最少消费。

题型 动态规划、最短路径

题解 令 $f[x_1, x_2, x_3, x_4, x_5]$ 表示每件物品的分别购买 $x_1 \sim x_5$ 件时的最低消费额, $f[x_1, x_2, x_3, x_4, x_5] = \min(f[x_1 - q[1], x_2 - q[2], x_3 - q[3], x_4 - q[4], x_5 - q[5]] + p[i])$, 也可将每种状态 $[x_1, x_2, x_3, x_4, x_5]$ 都作为一个顶点, 每项优惠则作为一条边, 求 $[0, 0, 0, 0, 0]$ 到目标状态的最短路。

3.3.3 Camelot (camelot)

题目 将棋盘中的国王和 N 个骑士移动到同一格，王每次可以走到相邻的 8 格之内，骑士则按“日”字前进，王可另由骑士护送，从他俩相遇的那个点开始一起行动，计算一共所需的最小步数。

题型 最短路径

题解 求出任意两个方格之间的最短路径，再枚举集合点，最终结果为骑士们到集合点的距离和，再加上王走到该点的距离，这里需分三种情况讨论：(1)王自行走到集合点；(2)骑士护送；(3)王前进一段距离后，再由骑士护送。

3.3.4 Home on the Range (range)

题目 一个由 0 和 1 组成的矩形，求其中所有由 1 组成的子正方形边长。

题型 动态规划、二维线段树*

题解 令 $f[i,j]$ 表示以 (i,j) 为右下角的可构成的最大的正方形边长，方程为 $f[i,j] := \min(f[i-1,j], f[i,j-1], f[i-1,j-1]) + 1$ ，由于本题需统计所有可能的正方形边长，最后应将长度不超过 $f[i,j]$ 的方案一并统计后输出。

3.3.5 A Game (game1)

题目 两人均按最优策略，轮流从一正整数序列的两端取数，求最后结果。

题型 博弈论、动态规划

题解 令 $f[i,j]$ 表示序列中第 i 至 j 个数所组成的子序列中选手 A 的最优取值结果， $f[i,j] = \max(f[i+1,j] + a[i], f[i,j-1] + a[j])$ ，选手 B 的结果即为 $\text{sum}[i,j] - f[i,j]$ 。

3.4.1 Closed Fences (fence4)

题目 一个由平面上一些不相交的线段首尾相连而围成闭合栅栏，给出一个参考点的坐标，求从参考点上所能看见的栅栏列表。

题型 计算几何

题解 用二分法确定线段上的取样点，若连接参考点和取样点的线段上没有其他相交或重叠线段，则该点及所在线段可见，但需注意精度和几种特殊的相交情况。

3.4.2 American Heritage (heritage)

题目 给出二叉树的先序遍历和中序遍历，求该树的后序遍历。

题型 递归

题解 根据二叉树遍历的性质，先序遍历的第一个结点必为当前子树的根，而在中序遍历中根的位置之前必为根的左子树，之后必为根的右子树，再递归建立其左右子树，直至完成整棵二叉树的建立，最后输出其后序遍历即可。

3.4.3 Electric Fence (fence9)

题目 有一个由三个给定点所围成的三角形，求三角形内所有的整数点。

题型 枚举、皮克定理

题解 枚举三角形内所有整点即可，也可以使用皮克定理：面积 S 等于内部整点数 a 与边上整点数 b 除 2 的商的和加 1，即 $S=a+b/2+1$ 。

3.4.4 Raucous Rockers (rockers)

题目 用 N 首歌灌制 M 张唱片，每张唱片总时长为 T ，在保持歌曲顺序不变的前提下，求最佳方案使得发行的歌曲总数尽量多。

题型 动态规划、记忆化搜索

题解 令 $f[i,j,k]$ 表示前 i 首歌装于 j 张唱片且第 j 张唱片剩余时间为 k 时最多可能的灌制的歌曲数，状态转移方程 $f[i,j,k]=\max(f[i-1,j,k], f[i-1,j,k+a[i]]+1, f[i-1,j-1,0]+1)$ 。

*Chapter4 Advanced algorithms and difficult drills***4.1.1 Beef McNuggets (nuggets)**

题目 已知 N 个数，求无法由这些数累加得到的最大正整数。

题型 动态规划、构造法

题解 令 $f[i]$ 表示正整数 i 可否由所给数字相加得到，可以证明结果不会超过最大两个数的最小公倍数， $f[i] = f[i] \text{ or } f[i - a[j]]$ ($1 \leq j \leq n$)，也可以尝试使用搜索直接构造。

4.1.2 Fence Rails (fence8)

题目 现有 N 段原木，需切割成 R 块木料，原木和木料的长度均不唯一，求最佳的切割方案以使木料需求尽可能得到满足。

题型 迭代加深

题解 迭代加深搜索可以切出的木料数，用二分法枚举深度，最终结果即是最大的合法深度。

4.1.3 Fence Loops (fence6)

题目 从无向图的任一点出发后再回到该点，求所经过的最短路径。

题型 最小环

题解 先需将题目中所给的边转化成点，再由每个顶点 i 开始求单源最短路，枚举其他所有顶点 j ，此时松弛前后的路径 (i, j) 便构成了一个环，求出环长的最小值即可。

4.1.4 Cryptcowgraphy (cryptcow)

题目 有一种加密方式，随机地把 C,O,W 三个字母插到原字符串中，通过交换 C-O 和 O-W 之间的子串实现加密，现给出加密后的信息，求此信息被加密的次数。

题型 搜索、字符串哈希

题解 通过搜索不断对加密字串进行解密，最后验证还原信息的合法性即可，但在搜索过程中需加入大量剪枝优化，其中最关键的有字符串哈希判重、可行性剪枝和优化搜索顺序。

4.2.1 Drainage Ditches (ditch)

题目 一套由单一源点向汇点连接的排水系统，求这套系统的最大容量。
题型 最大费用流
题解 经典的网络流模型，常用算法有 Ford-Fulkerson、Dinic 和 SAP。

4.2.2 The Perfect Stall (stall4)

题目 每头奶牛都有一些固定的产奶点，每个产奶点同时只能容纳一头奶牛，求最佳的匹配方案使得产奶量尽可能多。
题型 二分图匹配
题解 经典的最大匹配问题，可以使用网络流或匈牙利算法予以解决。

4.2.3 Job Processing (job)

题目 M_1 个 A 型机器对工件进行粗加工，之后再由 M_2 个 B 型机器对其进行精加工，每个机器加工单个工件的时长有所不同，求完成 N 个工件的粗、精加工各自所需最少耗时。
题型 贪心
题解 根据问题的单调性，使精加工耗时最短前提条件必然是粗加工耗时同样最短，可分别记录每台机器当前已预订工作时长和总耗时，并且始终将新的工件分配给预订时长与加工时长之和最小的 A_i 和 B_i 。

4.2.4 Cowcycles (cowcycle)

题目 在给定大小范围内找出两组数，使得前后两组数之商满足最大值超过最小值的 3 倍，并且排序后相邻两项商的差的方差最小。
题型 搜索
题解 搜索即可，但须根据题目限制予以剪枝。

4.3.1 Buy Low, Buy Lower (buylow)

题目 最长下降子序列问题，计算序列长度和不重复的方案数。
题型 动态规划
题解 在求解最长长度时，可令 $f[i]$ 表示截止到第 i 个数的最长下降子序列长度， $f[i] = \max(f[i-j]) + 1$ ($1 \leq j \leq n, a[j] > a[i]$)，另设 $g[i]$ 表示到第 i 个数为止的最长序列方案数， $g[i] = \sum (g[j])$ ($1 \leq j \leq i-1$)，记录下序列方案中的上一位置 $a[j]$ ，避免重复统计相同的方案。

4.3.2 The Primes (prime3)

题目 一个 5×5 的素数方阵，每行每列以及两条对角线上从上到下从左至右均由 5 位素数构成，并且这些素数五个数位上的和必须相等，现给出左上角的数和素数位数之和，计算所有可能的方案。

题型 搜索优化

题解 因为本题中所有方格相互牵制，故而优化搜索顺序就变得至关重要，可以枚举两条对角线，接着枚举中间一竖列，再是第一和第五行，最后枚举第二和第四行，此时第三行已由减法得到。

4.3.3 Street Race (race3)

题目 在一个有向有环图中，从顶点 0 走向顶点 N，求必定经过的顶点和能够将图分割为前后两个互不连通分量的中间顶点。

题型 搜索

题解 枚举删除非起始点，从起点深搜判断是否仍有通路，若无则该点为必经点，此时再继续搜索判断是否重复途径已经点，若否则该点为中间点。

4.3.4 Letter Game (lgame)

题目 已知字母与数值间的对应规则，现给出标准串和一些字符串，每两个字串之间可以组成字对，计算所有数值之和与标准串相同的字串或字对，并且其中每个字母的出现频数不能大于标准串中的频数。

题型 枚举

题解 预先排除掉所有非法字串，再依此生成所有合法字对，最后枚举验证每个字串和字对的合法性即可。

4.4.1 Shuttle Puzzle (shuttle)

题目 一排各有 N 个的黑白棋子，棋子每步可以移动到相邻或相间一颗棋子的空格，初始时左起 N 格为白子，右起 N 格为黑子，中间一格为空格，求调换所有黑白棋子位置的步数最少的移动方案。

题型 构造法、搜索

题解 本题可以用宽搜解决，然而通过对较小规模数据的观察，不难在本题移动方案的前 $n+1$ 个步骤所组成的序列中存在着明显的规律，而其后 n 个步骤也可由此对称得出。

4.4.2 Pollutant Control (milk6)

题目 一套由单一源点向汇点连通的运输网络，求代价最小的连边切割方案，使得源汇点之间不再连通。

题型 最小边割

题解 根据最大流最小割定理，最小边割的容量即等于最大流的流量。在求割边集时，可以先枚举去掉每条边后再求一次新图的最小割，若边的容量恰好等于流量的减少量，则该边必在解集内，称之为确定边。再用迭代加深求出割集内容量未满足的边，若这些边连同已确定边的流量总和与最小割值相等，并且去掉这些边后源汇点不再连通，则这些边也在解集内。

4.4.3 Frame Up (frameup)

题目 将若干个矩形方框按次放入平面中，方框之间相互重叠但始终有可见部分，现给出最终的俯视图，求方框的叠放顺序。

题型 拓扑排序

题解 由每个方框可见部分的最大和最小横纵坐标得出四个角的坐标，以此确定方框 i 的四条边，并在边上寻找覆盖 i 的其他方框 j ，对于每一个覆盖矩形建立一条有向边 (i, j) ，最后求出所有合法的拓扑排序序列即可。

Chapter5 Serious challenges

5.1.1 Fencing the Cows (fc)

题目 二维平面上有若干个点，求一个周长最小且符合条件的凸多边形，能够将所给点集完整包含。

题型 二维凸包

题解 典型的二维凸包问题，常用算法有 Graham 扫描法即卷包裹法。

5.1.2 Starry Night (starry)

题目 一个由 0 和 1 组成的二维矩阵，所有相连通的 1 组成一个单位图形，每个图形有八种形式的翻转变形，现在需要对矩阵中图形进行标记，相同图形应打上相同标记。

题型 搜索

题解 用灌水法找出单位图形，若之前这种图形没有出现过，则将其八种变形加入散列表中，否则将打上相同标记。

5.1.3 Musical Themes (theme)

题目 给出 N 个正整数，求其中最长的两个不相交的重复子序列。

题型 动态规划、字符串匹配

题解 先求出 N 个数的两两之差 $a[1..n-1]$ ，题目实际要求的即是 $a[i]$ 的重复长度，可令 $f[i,j]$ 表示分别以第 i 和 j 个数结尾的重复子序列长度，方程由最长公共子序列问题变形可得， $a[i] \neq a[j]$ 时 $f[i,j]=0$ ，否则 $f[i,j] = \begin{cases} f[i-1,j-1]+1 & (j-i > f[i-1,j-1]+1) \\ j-i-1 & (j-i \leq f[i-1,j-1]+1) \end{cases}$ ，但因空间有限，最后还需使用滚动数组优化，输出结果为 $\max(f[i,j])+1$ 。

5.2.1 Snail Trail (snail)

题目 在 $N \times N$ 的棋盘中有若干的障碍点，现从左上角开始沿直线前进，遇到障碍点或棋盘边缘则 90° 转向，直到遇到已经点才停止，求经过的最多格子数。

题型 搜索

题解 因障碍物总数不超过 26 个，直接深搜即可，每次沿直线前进，直至碰到障碍点、棋盘边缘或已经点。

5.2.2 Electric Fences (fence3)

题目 二维平面上有若干条平行于坐标轴的线段，求解平面内到所有线段两端或垂足的距离总和最短的点。

题型 搜索

题解 题目要求精确到小数点后一位，可将坐标轴整体放大 10 倍，从几何中心开始搜索，每次选择相邻四个方向上距离总和更短的点，可以迅速得到局部最优解，再多次搜索随机起点，重复上述操作即可求出全局最优解。

5.2.3 Wisconsin Squares (wissqu)

题目 牧场和仓库里分别有固定数目的 5 种奶牛，同种奶牛不能处于相邻位置，给出牧场上奶牛的分布情况，现需将在牧场和仓库的奶牛进行交换，求字典序最小的交换方案和总方案数。

题型 搜索

题解 直接深搜即可，预处理 $f[i,j,k]$ 表示与 (i,j) 相邻的第 K 种奶牛数，若 $f[i,j,k]=0$ 则表明当前位置可以放置该种奶牛。

5.3.1 Milk Measuring (milk4)

题目 商店里有 P 种容量不同的容器，同种容器可以重复购买，求一套购买种类数最少的方案，使得这些容器容积之和恰好为 Q 。

题型 迭代加深、动态规划

题解 迭代加深搜索购买方案，逐次加大购买种类数，注意重复购买同种容器，容量总和超过 Q 则跳出。本题也可使用动态规划求解，令 $f[i,j]$ 表示前 i 个容器容量和为 j 的最少种类数， $f[i,j]=\min(f[i-1,j], f[i-1,j-a[i]*k]+1)$ 。

5.3.2 Window Area (window)

题目 对二维平面中的矩形进行若干次放置、移除、调整叠放顺序等操作，所有矩形均平行于两坐标轴且带有标记，动态求解拥有同种标记的矩形的面积可见率。

题型 矩形切割

题解 本题矩形数目和坐标范围都很大，可以使用矩形切割进行求解，并用双向链表记录叠放顺序，模拟五种操作即可。

5.3.3 Network of Schools (schlnet)

题目 一个连通的有向无环图，求遍历全图所需最少的起点数，以及最优的连边方案使得全图强连通。

题型 强连通分量

题解 求出图中所有极大强连通分量，将每个分量都缩为一个点，遍历起点即为入度为 0 的顶点，再在所有入度为 0 和出度为 0 的顶点间连接新边即可使全图强连通。

5.3.4 Big Barn (bigbrn)

题目 一个有若干损坏区域的矩形，求其中最大的完整子正方形的边长。

题型 动态规划

题解 状态转移方程同 3.3.4 range 一题，可令 $f[i,j]$ 表示以 (i,j) 为右下角的可构成的最大的正方形边长， $f[i,j] := \min(f[i-1,j], f[i,j-1], f[i-1,j-1]) + 1$ 。

5.4.1 All Latin Squares (latin)

题目 在 $N \times N$ 的数字棋盘中，前 N 个自然数在每行每列均出现且只出现一次，第一横行已经确定，求合法的摆放方案总数。

题型 搜索

题解 搜索构造合法方案，因第一行已经确定，故只需从 $(2,2)$ 开始搜索剩下边长为 $n-1$ 的矩形，实际上此过程中第一列剩余 $n-1$ 个数也被确定，总方案数即当前结果乘以 $(n-1)!$ ，但本题还需使用置换思想优化，若出现相同置换圈，顺序交换后情况等价，则不必深入搜索，只需累加方案数即可。

5.4.2 Canada Tour (tour)

题目 在一个顶点有序的无向图中，按单调顺序从第一个顶点出发到达最后一个顶点后再返回起始点，除起始点外其他顶点不可重复经过，求一套最佳的访问方案使得经过的顶点尽可能多。

题型 动态规划、网络流

题解 问题等同于甲乙两人同时从起点出发，经过不同的顶点后到达终点，可令 $f[i,j]$ 表示甲乙分别到达顶点 i 和 j 时所经过的最多的顶点数，方程 $f[i,j] = \max(f[i,1..j-1]) + 1$ 。

5.4.3 Character Recognition (charrec)

题目 26 个英文字母和空格的字体模型，分别用 27 个 20×20 的 01 点阵表示，现给出一段已损坏的字符图案，可能会有 01 点阵错误，也可能丢失或多余一行，求对应的原字符信息。

题型 动态规划

题解 每段图案必有差异率最小的标准字体与之对应，可用 $g[i,j,k]$ 统计图案中第 i 行与 j 号字体模型的第 k 行的误差率，再令 $f[i,j]$ 表示图案中第 i 行及其后 j 行的最小误差之和，按 $j=19 \sim 21$ 三种情况枚举丢失或多余的行序，最后全图误差总和即为 $dp[i]=\min(dp[i-19], dp[i-20], dp[i-21])$ ，转移的同时记录输出方案便可。

5.4.4 Betsy's Tour (betsy)

题目 在 $N \times N$ 的棋盘中，从左上角开始到左下角为止遍历整个棋盘，求可能的方案总数。

题型 搜索、动态规划

题解 可以使用基于连通性状态压缩的动态规划或搜索剪枝解决，主要依靠可行性剪枝，若某点周围只有一个未经点则该点为必经点，若当前点周围有两个及以上非终点的必经点，则势必会形成死胡同。

5.4.5 TeleCowmunication (telecow)

题目 一套由单一源点向汇点有向连通的电脑网络，求代价最小的顶点切割方案，使得源汇点之间不再连通。

题型 最小点割

题解 可以用拆点法将最小点割转化为最小边割问题，使得最小点割的割值即为新构造的流网络中的最大流值，最后依次枚举删除顶点，若删去后最大流值比原流值少 1，则该点必为割点。

5.5.1 Picture (picture)

题目 N 个矩形相互平行地重叠摆放在二维平面中，求重叠图形的周长。

题型 线段树

题解 将横纵边分别排序，坐标相同时始边较终边靠前，再按序枚举横纵边，遇始边时将其在坐标轴上投影部分的附加值加 1，遇终边减 1，若某一时刻附加值增至 1 或减至 0，则将该段投影长度计入周长。

5.5.2 Hidden Passwords (hidden)

题目 在 L 个字符组成的字串中，字串头尾可循环衔接，现需找出其中长度为 L 且字典序最小的子串。

题型 枚举

题解 用 min 和 now 两个指针进行比较，分别初始化为 1 和 2，令 now 指针向后循环推进，每次分别枚举两指针其后 $L-1$ 个字符，若 $s[now+i] < s[min+i]$ 则两指针均跳至 $now+i-1$ ，若 $s[now+i] > s[min+i]$ 则 now 指针跳至 $now+i-1$ ，以此避免相同前缀的重复比较。

5.5.3 Two Five (twofive)

题目 在 5×5 的棋盘放入前 25 个字母，每行每列都严格递增，现给出摆放方案求方案序号，或给出方案序号求摆放方案。

题型 记忆化搜索、康拓展开

题解 记忆化搜索，可令 $f[a,b,c,d,e]$ 表示将前 $a+b+c+d+e$ 个字母放入棋盘中且第 1 至 5 行分别摆放 a 至 e 个字母时的方案总数，累加所有序号更小的方案数即可求出已知方案的序号，类似地逐位累计方案数便可求出已知序号的对应方案。

Chapter6 Contest Practice

6.1.1 Postal Vans (vans)

题目 在 $N \times 4$ 的棋盘中，从左上角出发沿棋盘线遍历所有顶点后再返回起点，计算合法方案的总数。

题型 动态规划、递推

题解 由于答案范围过大，需使用高精度运算，当 $n \geq 5$ 时存在如下递推关系 $f[n] = 2f[n-1] + 2f[n-2] - 2f[n-3] + f[n-4]$ 。

6.1.2 A Rectangular Barn (rectbarn)

题目 一个有若干块损坏区域的矩形，求其中最大的完整子矩形的边长。

题型 动态规划

题解 令 $h[i,j]$ 、 $l[i,j]$ 、 $r[i,j]$ 分别表示 (i,j) 位置向上左右三个方向最多可延伸的距离，每行状态只与上一行有关，可以使用滚动数组优化空间。

$$h[i,j] = \begin{cases} h[i,j-1] + 1 & \text{未损坏} \\ 0 & \text{已损坏} \end{cases}$$

$$l[i,j] = \begin{cases} \min(l[i,j-1], tl) & \text{未损坏} \\ +\infty & \text{已损坏} \end{cases}$$

$$r[i,j] = \begin{cases} \min(r[i,j-1], tr) & \text{未损坏} \\ +\infty & \text{已损坏} \end{cases}$$

其中 tl 、 tr 为当前点到左右两侧最近一处坏点或棋盘边缘的距离。

6.1.3 Cow XOR (cowxor)

题目 一串二进制长度不超过 21 的自然数，令 $sum[i]$ 表示前 i 个数连续异或的结果，求 $sum[i]$ 异或 $sum[j]$ 的最大值和 i,j 的位置 ($1 \leq i,j \leq n$)。

题型 位运算

题解 枚举所有的 sum 数，用 $f[0..4194303]$ 记录到目前为止所有出现过的二进制前缀，由此求出当前数与之前所有数所能取得的最大异或结果，同时记录所有数中最优结果的位置 t ，向前逆向枚举 $sum[j]$ ，当 $sum[t]$ 异或 $sum[j]$ 结果最优时即可得到答案。

附：参考资料

多年来 USACO 题库在国内诸多前辈的不断研习精深下，留下了大量的解题报告和丰富的经验总结，成为了我们今天竞赛学习时的一笔不可多得的财富。笔者在解题和总结过程中也屡有参考网络资料，以下予以简单列述，未能详尽标明题解出处者还望见谅。

原创题解	
Leokan	http://hi.baidu.com/leokan
北极天南星	http://starforever.blog.hexun.com
cuitianyi	http://cuitianyi.com
billylinux	http://billylinux.blog.hexun.com
中文译题	
NOCOW	http://www.nocow.cn
WZOI	http://www.wzoi.org/usaco