|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 题目大意 | 算法讨论 | 时间 | 空间 |
| 2013 F | Low Power | 将2nk个电池放在n个机器的2n个芯片上，每个芯片的能量石芯片上k个电池的能量的最小值，使得n个机器芯片的能量值之差的最大值最小。 | 先二分答案是多少，再贪心选择电池，在满足两个芯片的差不超过答案的情况下尽量选择能量值小的电池作为两个芯片的能量的最小值。 | O(NlogN) | O(N) |
| 2008 B | Always an Integer | 给一个整系数多项式，问当自变量取任意正整数时多项式模D是否恒为0。 | 计算f(n+1)-f(n)，这是一个E-1次多项式，现在只需要计算f(0)是否被D整除和E-1次多项式f(n+1)-f(n)对于任意正整数n是否被D整除。 | O(E^3) | O(E^2) |
| 2003 C | Riding the Bus | 给定N阶SZ曲线的定义，问N阶SZ曲线上某一段曲线的长度是多少。 | N阶SZ曲线由9个N-1阶SZ曲线组成，先计算出两个端点属于哪个N-1阶SZ曲线，再递归处理。 | O(2^N) | O(N) |
| 1999 A | Bee Breeding | 数字在蜂巢中按螺旋排列，问两个数字对应的蜂巢中的格子的距离是多少。 | 可以把蜂巢拆成以1为中心的若干个圈，第i个圈包含的数字个数为6i，可以快速计算出每个格子相邻的6个格子的编号，然后Bfs一遍就可以了。 | O(N) | O(1) |
| 1998 E | Petri Net Simulation | 一个Petri网有若干库所和若干变迁，库所可以存储令牌，一个变迁有一些输入库所和一些输出库所，每次变迁导致每个输入库所的令牌输减一，输出库所加一，模拟若干次变迁后的结果。 | 直接模拟。 | O(N\*NF) | O(N) |
| 1999 C | A Dicey Problem | 一个骰子在一张图上滚动，每次只能滚动到与顶面数字相同的格子上，求从起点再滚回起点的路径。 | 搜索，记录状态f[x][y][t][f]，表示当前骰子的位置是(x, y)，顶面数字是t，前面数字是f。 | O(NM\*6^2) | O(NM) |
| 1999 E | Trade on Verweggistan | 有若干个工厂，每个工厂有从上到下摆着若干堆货物，必须先买上面的货物才能买下面的货物，问最大的利润和购买的货物数量。 | 先求出每个工厂货物的前缀和，然后用f[i][j]表示前i个工厂买j个货物能否获得最大的利润，枚举当前工厂购买多少货物转移即可。 | O(w^3b) | O(w^2b) |
| 2000 E | Internet Bandwidth | 求一个无向网络的带宽。 | 网络流。 | O(N^2M) | O(M) |
| 2000 A | Abbott’s Revenge | 求箭头迷宫中从起点走到终点的最短路径。 | 记录状态f[x][y][d]，位置为(x, y)，朝向d，然后Bfs。 | O(9^2\*4T) | O(9^2\*4) |
| 2001 F | A Major Problem | 求音符从一个音阶转到另一个音阶后是什么音符。 | 打表打出每个音阶的音符，然后直接输出。 | O(15\*7) | O(15\*7) |
| 2001 B | Say Cheese | 一个虫子在奶酪中从一个点爬到另一个点，在奶酪的球体中爬行不花费时间，求最短时间。 | 最短路径。 | O(N^2) | O(N) |
| 2002 C | Crossing the Desert | 求从沙漠中的起点走到终点需要带的最少的食物。每走过一个单位消耗小单位水和一单位食物，沙漠中有若干个绿洲，经过绿洲可以补充水和储藏食物，携带的水和食物的总量不能超过一个定值。 | 从终点开始，逆向SPFA。 | O(N^2) | O(N) |
| 2002 A | Balloons in a Box | 一个盒子中有若干个点，可以以一定顺序在每个点上放气球，就是以这个点为球心，让这个求膨胀，直到碰到盒子边界或其它气球，求盒子中不被气球占据的最小体积。 | 求1~n的全排列，然后模拟。 | O(N!N^2) | O(N) |
| 2003 H | A Spy in the Metro | 有若干趟列车，从起点开向终点或从终点开向起点，问从起点出发在指定的时间到达终点在站台上等待的时间最小是多少。 | f[i][j]表示第j个时刻在第i个站台的最小等待时间，f[i][j] = min(f[i][j – 1] + 1, f[i – 1][j – t[i – 1]], f[i + 1][j – t[j]] )。 | O(NT) | O(NT) |
| 2003 B | Light Bulbs | 有N个灯泡，N个开关，每个开关控制相邻三个灯泡，问从初始状态到目标状态要按动哪些开关。 | 先计算初始状态与目标状态的异或值V，如果Vi = Vi+1，则第i-1个开关和第i+2个开关必须同时开启或关闭，否则这两个开关的状态必须不同。所以确定前3个开关的状态就可以确定所有开关的状态，再判断是否可行。 | O(N^2) | O(N) |
| 2004 I | Suspense! | 两栋楼之间距离为d，两个人分别住在两栋楼的第j、t层中，要从这两层窗户下端搭一座桥，使得桥的主缆尽量长且猫不能通过桥抓到鸟。 | 桥的高度只可能与某层的窗户下端高度相同或比窗户下端高出0.5m。从下到上枚举每种高度是否可行，然后用抛物线长度公式计算结果即可。由于数据组数非常大，读入时每次读入一行即可通过全部数据。 | O(N) | O(N) |
| 2004 H | Tree-Lined Streets | 在若干条大街上种树，每棵树至少与路口相距25m，两棵树至少相距50m，问最多种多少棵树。 | 求每条大街与其他大街的交点，这条大街被路口分成了若干段，求每一段最多种多少棵树。 | O(N^2) | O(N) |
| 2004 G | Navigation | 有若干个信号源，每个信号源以一定速度向一定方向运动，到达指定时间目标点会向信号源发送信号，已知每个信号源收到信号的时间，求目标点的位置。 | 先求出每个信号源到达指定时间时所在的位置P，再计算信号到达信号源需要经过的路程r，则目标点一定在以P为圆心，r为半径的圆上，然后计算N个圆的交点即可。 | O(N) | O(N) |
| 2005 J | Zones | 从N个圆中选择K个圆，使得这K个圆覆盖的人数最多，输出覆盖人数最多的方案中字典序最小的。 | 枚举选择哪N个圆，再计算这N个圆覆盖了多少人。 | O(2^N\*N) | O(N) |
| 2005 I | Workshops | 有N个讨论会，每个讨论会由一定人数参加，持续一定时间，有M个房间，每个房间能容纳一定人数，每个房间的使用时间有限制，问最少有多少个讨论会不能在房间中举行，以及在上面的条件下不能在房间中举行讨论会的最少的人数。 | 将每个房间的使用时间从小到大排序，依次考虑，每次选择这N个讨论会中能在这个房间举办且人数最多的讨论会在这个房间中举办，这样贪心选择即为最优方案。 | O(NM) | O(N+M) |
| 2005 G | Tiling the Plane | 问能否用给定的多边形铺满平面。 | 预处理从任意两个点开始的边界最多有多长是相同的，然后枚举边界连接点即可。 | O(N^3) | O(N^2) |
| 2005 E | Lots of Sunlight | 有若干栋楼，每次询问一栋楼中的一层被太阳直射的时间段。 | 对于每个询问，计算每栋楼的楼顶与询问的楼层的低端的连线与x轴的交点，然后用反三角函数计算出这条线与x轴的夹角，再折算成时间。 | O(NQ) | O(N) |
| 2005 C | The Traveling Judges Problem | 一张图上有N个点，有P个人从P个地点走到目的地，问这P个人走过的路径的总长度最小的方案。 | f[S][i]表示集合S的人走到目的地i的最小长度，f[S][i] = min(f[A][j] + f[A^S][k] + dist[j][i] + dist[k][i])，其中A是S的非空子集，j和k是是两个点。这个动态规划的复杂度可以优化到O(3^P\*N)。 | O(3^P\*N) | O(2^P\*N) |
| 1998 G | Spatial Structures | 求一个图的四分树或一个四分树对应的图。 | 递归处理。 | O(N^3) | O(N^2) |
| 2013 A | Self-Assembly | 一个分子有四个表面，每个表面有字母和符号，如果两个分子的两个表面字母相同符号相反，那么这两个分子可以由这两个表面连接在一起，问这些分子能否连接成无限大的结构。 | 把每种表面视为一个点，一共52个点，对于每个分子上任意两个表面，从一个表面的符号相反的表面向另一个表面连一条有向边，然后判断这个图是否有环即可。 | O(N) | O(52\*52) |
| 2013 C | Surely You Congest | 有C个人从各自的起点同时走向终点1，每人只会走最短路，不能有两个人同时经过一条路，问最多有多少人能走到终点。 | 先求出最短路图，然后按C个人中距离终点相等的分成一类，求网络流。 | O(CM) | O(M) |
| 2013 H | Матрёшка | 把N个并排的套娃合并成若干个完整的套娃，每次只能合并相邻两个套娃，问最少拆开多少次套娃。 | 合并两个套娃的代价是套娃的总数减去一个套娃比另一个套娃的最小的套娃小的个数的最大值，预处理每个区间比每个值小的套娃个数，然后dp即可。 | O(N^3) | O(N^3) |
| 2012 B | Curvy Little Bottles | 给一个N次多项式，把这个多项式的图像从x=low到x=high截取，然后绕x轴旋转一周形成瓶子，问瓶子的体积和等体积标记的位置。 | 用微积分求出瓶子体积关于x的函数，然后二分即可。 | O(N^2) | O(N) |
| 2012 C | Bus Tour | 求从起点经过经过所有旅馆到达终点再从终点经过所有旅馆到达起点的路程的最小值，满足从起点出发到达的前一半的旅馆必须是从终点回来到达的前一半的旅馆。 | 分别从起点和终点状压dp，f[S][i]表示经过的点的集合为S，当前到达点i的最小路程，然后再枚举那一半的旅馆的集合，计算所有可能的集合的最小值即可。 | O(2^N\*N^2) | O(2^N\*N) |
| 2012 E | Infiltration | 有N个人，任意两个人A、B要么A控制B要么B控制A，控制了一个人A就可以控制A直接控制的所有人，问最少控制多少人。 | 迭代加深搜索。 | O(N^6) | O(N^2) |
| 2013 E | Harvard | 有b个内存库，每个内存库可以存放s个变量，每次操作可以访问第0个内存库，修改BSR的值或访问第BSR个内存库。给定一个程序，包含若干个变量访问和一些循环，把每个变量对应到一个内存库中，使得总操作次数最小。 | 先枚举第0个内存库有哪些变量，然后计算出这个程序包含了多少个0号内存库变量，以及其它内存库变量Vi变到Vj的总次数，然后搜索每个变量属于哪个内存库，如果Vi变到Vj属于两个不同的内存库，对答案的贡献为2，否则为1，最后加一些剪枝即可。 | O(2^b\*b^3) | O(Nb^2) |
| 2012 D | Fibonacci Words | 问模式串在斐波那契串中出现的次数。 | 模式串在F(N)中出现的次数等于F(N-2)的次数加上F(N-1)的次数再加上F(N-2)和F(N-1)连接处出现的次数，这样只需要计算F(i)的前L个字符和后L个字符即可。 | O(NL) | O(L) |
| 2013 D | Factors | 求最小的正整数k，使得k的质因子的排列方案数等于n。 | 先枚举k的质因子个数，质因子个数的阶乘再除以每个质因子指数的阶乘要等于n，搜索每个质因子的指数，若质因子a < b则a的指数一定大于等于b的指数，再加一些剪枝即可。 | O(?) | O(64\*18) |
| 1998 D | Page Selection by Keyword Matching | 每个网页有一些关键词，每次查询一些关键词，输出最相关的5个网页。 | 直接处理即可，注意匹配不区分大小写。 | O(25\*20\*8\*8\*Q) | O(25\*20\*8) |
| 1998 B | Flight Planning | 一次飞行有K个航段，每个航段两万米和四万米的风速，风速在两万米到四万米间是线性的，求每个航段的飞行高度，使得总耗油最小。 | f[i][j]表示后i个航段，第i-1个航段的飞行高度为j的最小耗油，转移即可。 | O(20^2\*K) | O(20\*K) |
| 1999 D | The Fortified Forest | 有若干棵树，每颗树有自己的坐标、价值和砍掉它能建造的栅栏长度，求最少砍掉多少价值的树使得剩下的树用栅栏围住。 | 暴力枚举砍掉哪些树，然后求凸包。 | O(2^N\*NlogN) | O(N) |
| 2000 B | According to Bartjens | 给n个数字，要在这个n个数字中插入加号、减号或乘号使得运算结果为2000。 | 暴力搜索每种可能再计算即可。 | O(4^N) | O(ans\*N) |
| 2000 C | Cutting Chains | 有n个链环，初始时有一些链环连接到了一起，问至少拆多少个链环可以把n个链环连接成一条链。 | 枚举拆掉哪些链环，然后判断没被拆掉的链环是否是由若干个链组成的，以及链的个数是否小于等于拆掉的链环的个数加一即可。 | O(2^N\*N^2) | O(N^2) |
| 2000 F | Page Hopping | 求任意两点最短路径的平均值。 | Floyd求任意两点最短路。 | O(N^3) | O(N^2) |
| 2001 A | Airport Configuration | 机场中有N个到达门，N个出发们，每个门对应着一个城市，从每个到达城市到每个出发城市有一些客流，问所有人走过的距离总和。 | 直接计算。 | O(N^2) | O(N^2) |
| 2001 H | Professor Monotonic's Network | 问包含K个比较节点的比较网络是否是排序网络，以及这个比较网络的运行时间（比较节点可以并行计算）。 | 随机生成足够多次（我用了200000次）1~n的排列，然后检查计算完的序列是否是1~n。 | O(200000\*K) | O(K) |
| 2001 G | Fixed Partition Memory Management | N个程序，M个存储器分区，每个程序在每个存储器的分区运行要花费一些时间，同一时刻一个存储器只能运行一个程序，求所有程序平均结束时间的最小值。 | 费用流，把每个存储器拆成N个点，第i个点表示这个存储器倒数第i个运行的程序，从每个程序到每个存储器拆的点连一条边，费用为这个程序在这个存储器中运行花的时间乘以i。 | O(N^3M) | O(N^2M) |
| 2002 H | Silly Sort | 有n个数，每次可以交换两个数，代价为两数之和，问把n个数变成升序的最小代价。 | n个数可以组成一些置换，在每个置换内，要么是用这个置换里的最小值不断地交换其它值，要么先把这个最小值与全局最小值交换，用全局最小值交换置换里其他值，再换回来。 | O(N) | O(N) |
| 2002 E | Island Hopping | 有N个岛屿，要建立一些电缆使得这N个岛屿连通，使得电缆总长度最小。每个电缆同时开始修建，问这个方案中所有人的平均等待时间。 | 最小生成树。 | O(N^2logN) | O(N^2) |
| 2002 G | Partitions | 如果把一个矩形划分成了若干个子矩形，则这是一种矩形的划分。给两种矩形的划分，问比这两种划分都精细的最粗糙的划分，以及比这两种划分都粗糙的最精细的划分。 | 先求两个划分的并，再求两个划分的交即可。但注意两个划分的交不一定是一种有效的矩形划分，如果不存在一条连接着左边界到右边界或上边界到下边界的直线，则一定无效，否则递归处理。 | O(w^2h^2) | O(wh) |
| 2003 A | Building Bridges | N\*M的网格，有些网格是建筑，要建造一些道路，每条道路必须是直的，连接着两个建筑且在网格边界上，问使得最多的建筑连通，至少建造多少条道路以及最小总长度。 | 枚举在每行和每列建造道路可以连接的建筑以及道路的长度，然后求最小生成树即可。 | O(NM(N+M)) | O(NM(N+M)) |
| 2003 J | Toll | 把p个货物从起点运到终点，经过村庄要收取1个货物，经过城镇每20个货物收取1个货物，问起点时至少带多少货物。 | 从终点开始SPFA。 | O(52^2) | O(52^2) |
| 2003 D | Eurodiffusion | 10\*10的格点上，有若干个国家，每个国家是一个矩形，国家没有公共点，且所有国家是连通的。一开始每个国家有一百万个自己国家的硬币，然后每天会向周围的城镇送硬币，每满1000个送一个，问每个国家什么时候可以同时拥有所有国家的硬币。 | 按每个国家依次模拟这个国家的硬币在什么时刻遍及到其它国家，再取所有国家的最大值。 | O(10000\*10\*10\*N) | O(10\*10) |
| 2003 F | Combining Images | 将一个图像编码，如果图像的所有像素都相同，编码以1开始，跟着像素的颜色，否则以0开始，然后是4个子图像的编码。给两个图像的编码，求两个图像的交集的编码。 | 如果两个编码都以1开始，那么直接求交，如果一个是以1开始的，那么要么返回0，要么返回另一个图像，否则递归处理四个子图像。注意如果四个子图像全为0，那么也应该返回0。 | O(N^2) | O(N) |
| 2013 J | Pollution Solution | 求圆与多边形的交集的面积。 | 依次考虑多边形的每条边，求两种图形的交，被切割的圆的面积可以分成一个三角形和一个扇形来求。 | O(N) | O(N) |
| 2004 C | Image is Everything | 一个物体是N\*N\*N的立方体，有一些单位立方体缺失了，每个单位立方体可以涂一种颜色。给这个物体的6个面的视图，问这个立方体最多由多少个单位立方体组成。 | 先假设这个物体不缺失单位立方体，然后遍历6个视图，将视图中的颜色涂到对应的立方体上，如果这个立方体已经被涂过了，则删掉这个立方体。 | O(N^4) | O(N^3) |
| 2004 E | Intersecting Dates | 有N个已经获得的日期区间，现在要知道M个日期区间，问还需要获得哪些日期区间。 | 把每个日期区间拆成左端点和右端点，把这些端点排序，然后按顺序扫描得出区间。 | O(N+M) | O(N+M) |
| 2005 H | The Great Wall Game | 在N\*N的网格上有N个棋子，求把这N个棋子移动成一行，一列或者对角线的最小步数。 | 如果移动成一行或者一列，直接按纵坐标或横坐标排序，然后贪心，如果移动成对角线，用KM算法即可。 | O(N^3) | O(N^2) |
| 2005 B | Simplified GSM Network | 有B个BST塔，C个电话机，每个电话机连接到最近的BST塔，有Q次询问，询问两个电话机之间要至少转换多少次BST塔。 | 求V图，然后floyd即可。 | O(B^3+C^3) | O(C^2) |
| 2005 F | Crossing Streets | 给起点和终点，和N条街，问从起点到终点至少经过多少条街。 | 先离散化，然后用双向队列BFS即可。 | O(N^2) | O(N^2) |
| 2006 A | Low Cost Air Travel | 有N条航线，每条航线有若干站点，乘坐这条航线要花费一定价钱，并且只能从航线的第一个站点做到后面的站点，有M个询问，每个询问给出一条路线，依次经过路线中的城市最少花费多少钱。 | f[i][j]表示已经走完了前i个路线中的城市，且当前在第j个城市，最少花费的钱数，按SPFA转移状态即可。 | O(N^5M) | O(N^3) |
| 2006 I | Degrees of Separation | 有N个人，有一些人有关系，问两个人需要经过的最小关系数的最大值。 | floyd。 | O(N^3) | O(N^2) |
| 2006 B | Remember the A La Mode | 有P种饼干，I中冰激凌，已知每种的个数，以及每种饼干配每种冰激凌的收益，求把饼干和冰激凌都用完的最小收益和最大收益。 | 费用流。 | O(P^2I^2) | O(PI) |
| 2006 E | Bit Compressor | 如果原串中存在连续若干个1，可以把这些1替换成1的个数，替换后必须长度更小，给定压缩后的串问原串是否存在或唯一。 | 枚举每种可能即可。 | O(2^20) | O(40) |
| 2007 A | Consanguine Calculations | 给出父母的血型或父母中一个人的血型和儿子的血型，求另一个人的血型。 | 枚举。 | O(1) | O(1) |
| 2007 G | Network | 有N个信息，每个信息会被分成一些信息块，一共有M个信息块，这些信息块会以一定顺序接收，接收时可以存进缓冲区或直接输出，每个信息输出必须是按顺序且连续的，问缓冲区大小的最小值。 | 枚举信息的接受顺序，然后模拟。 | O(N!NM) | O(NM) |
| 2007 F | Marble Game | N\*N的木板上放着M个小球，木板上还有一些分隔板，每次可以提起木板的一段，使所有小球向另一端滚动，要使每个小球都滚到对应的洞中，最少需要多少步。 | Bfs，用一个hash表判重即可。 | O(M^(N^2)) | O(N^2) |
| 2008 A | Air Conditioning Machinery | 在一个由若干小立方体组成的大立方体上，有一个入口和一个出口，要用不超过6个L型管道连接入口和出口，问最少用几个管道。 | 迭代加深搜索。 | O(8^6) | O(20\*20\*20) |
| 2007 E | Collecting Luggage | 有一条多边形行李带，行李在上面匀速运行，问一个人要拿到行李最少的时间。 | 二分时间，枚举两个顶点连成的线段，如果没有经过多边形内部则在这两个点间连一条边，然后求最短路。 | O(N^3) | O(N^2) |
| 2008 F | Glenbow Museum | 问长度为N的直角多边形序列有多少种符合要求。 | 序列中R的个数必须为N/2-2，不能有2个R相邻，DP即可。 | O(N^2) | O(N^2) |
| 2012 L | Takeover Wars | 两个公司，一个有N个子公司，一个有M个子公司，每次可以合并两个子公司或删除权值比它小的子公司，问哪个公司获胜。 | 每次操作如果这个公司的最大权值比另一个公司的最大权值大，就一定删除另一个公司的最大权值，否则合并这公司权值最大的两个子公司。 | O(N+M) | O(N+M) |
| 2012 K | Stacking Plates | 有N堆盘子，每堆盘子有Hi个盘子，下面的盘子比上面的盘子大。每次操作可以把一堆盘子的上面一部分移下变成新的一堆，或者合并两堆盘子使得下面的盘子比上面的盘子大，问合成一堆盘子的最小步数。 | 如果确定了最后的那堆盘子中，每个盘子属于哪个盘子堆，设一共有M块相邻盘子属于相同的盘子堆，则答案为2M-N-1。现在只需要确定大小相同的盘子属于哪个盘子堆，用f[i][j]表示大小为i的盘子的最上面的盘子属于第j个盘子堆，转移即可。 | O(NH^3) | O(NH^2) |
| 2008 J | The Sky is the Limit | 有N座山，每座山是一个底边在x轴上的等腰三角形，问这些山的轮廓的长度。 | 枚举这些山的所有交点，将这些交点的x坐标排序，对于每个x坐标，求出在这个点上的山最高的高度。 | O(N^3) | O(N^2) |
| 2008 I | Password Suspects | 密码有N位，由小写字母组成，有M个字符串在密码中出现，问可能的密码的个数。 | 先建立AC自动机，然后用状态f[i][j][S]表示密码前i位，当前在第j个AC自动机节点，已经出现的字符串集合为S。 | O(2^MNM) | O(2^MNM) |
| 2009 A | A Careful Approach | 有N架飞机，每架飞机有一个降落时间区间，要安排每架飞机的降落时间，使得最小的降落间隔最大。 | 先二分降落间隔，然后枚举N架飞机降落的先后顺序。 | O(N!) | O(N) |
| 2009 B | My Bad | 有一个逻辑电路，有N个输入，有与门、或门和异或门，最多有一个门会坏掉，可能永远输出0或1，或者永远与正确输出相反，给了一些实验数据的输入和输出，问哪个门以什么方式坏掉。 | 枚举，模拟。 | O(2^N) | O(N) |
| 2009 F | Deer-Proof Fence | 有N棵树，要建一些栅栏，使得每棵树离栅栏的距离都大于等于M，问栅栏的最小距离。 | 枚举每种点集，求包围这个点集的最小栅栏长度，即先求凸包，顶点用圆弧连接。然后用f[S]表示还剩集合为S的树没被包围，转移即可。 | O((2^N)^2) | O(2^N) |
| 2010 C | Tracking Bio-bots | N\*M的网格，有w座墙，只能往右或上走，有多少格子能到达右上角。 | 离散化，然后Bfs。 | O(w^2) | O(w^2) |
| 2009 H | The Ministers' Major Mess | 有N个议案，M个大臣，每个大臣可以同意或否决最多4个议案，每个大臣要有大于一半的建议被满足，问是否存在一种方案以及哪些议案被确定了。 | 建立2-SAT模型，如果不多于2个建议，则都要被满足，否则最多有一个不被满足。枚举每个议案通过或者否决，判断是否还有可行解。 | O(N^2M) | O(N^2) |
| 2010 D | Castles | 有N个城堡，N-1条路径，攻占一个城堡至少需要一些士兵，会消耗一些士兵，问至少多少士兵攻占所有城堡。 | 枚举士兵开始时在哪个城堡，然后树形dp，按每个子树需要的士兵减去消耗的士兵排序，再算出当前子树至少需要多少士兵。 | O(N^2) | O(N) |
| 2010 F | Contour Mapping | N行M列的等边三角带，每个顶点有一个高度，每隔h高度画一条等高线，问所有等高线的长度之和。 | 先枚举每条边是否是等高线，再枚举每个三角形，一个三角形中的所有等高线是呈等差数列的，快速计算即可。 | O(NM) | O(NM) |
| 2010 G | The Islands | 有N个岛屿，要从最左边的岛屿开始，只能向右走，经过一些岛屿到达最右边的岛屿，然后只能向左走，经过剩下的岛屿回到起点，两条路径必须分别经过两个点，求最短路径。 | f[i][j] (i<j) 表示第一条路走到了第i个岛屿，第二条路走到了第j个岛屿，这两条路径要同时到达最左边岛屿的最短路径。枚举第i-1个岛屿是第一条路还是第二条路转移即可。 | O(N^2) | O(N^2) |
| 2010 J | Sharing Chocolate | 有一个W\*H的巧克力，每次可以竖着或者横着掰成两块，问能否掰成N块指定大小的巧克力。 | f[S][i]表示较小的一边长为i的巧克力能否掰成S。预处理每种大小的巧克力可能由哪些集合组成，状态转移时枚举两块的大小，然后用预处理的信息转移。 | O(2^NWH) | O(2^NW) |
| 2011 H | Mining Your Own Business | N条边的连通图，要在一些点上装竖井，使得任意一个点被删除后，其它点都能到达一个竖井，问最少的竖井数和方案数。 | 先求割点，这些割点把原图分成了若干块，在每个只连有一个割点的块上安装竖井即可。如果没有割点，则在任意位置安装两个竖井。 | O(N) | O(N) |
| 2011 K | Trash Removal | N个顶点的多边形，要把这个这个多边形从一个固定宽度的管道扔下去，问宽度最小是多少。 | 枚举多边形的任意两个点，计算其它点到这两个点的线段的距离，计算在线段上面的最大距离与线段下面的最大距离之和，最小值即为答案。 | O(N^3) | O(N) |
| 2011 C | Ancient Messages | N\*M的图像有若干个黑色连通块，每个连通块可能包含一些洞，求出每个连通块包含几个洞。 | 先对每个不同的黑色连通块标记不同的数字，然后对于每个白色连通块，判断是否是一个洞，如果是一个洞就把包围它的黑色连通块加一。 | O(NM) | O(NM) |