ACM/ICPC World Final 试题泛做

姓名：章玄润

试题编号	名称	题目大意	算法讨论	时空复杂度
1999A	Bee Breeding	给定一张六连通的地图，每个格子均为一个正六边形，按照图中给定的方式标号，求某两个格子间的最小距离，相邻两点距离为 1。给定的两个格子的编号<=10000	任取三条过 1 号点直线中的两条，将一条设为 x 轴另一条设为 y 轴，然后模拟出每个标号的坐标值，每次移动可能是 x±1，y±1 或者 x 和 y 都发生变化，根据选取直线的不同分象限讨论即可。	时间复杂度：O(a) 空间复杂度：O(a)
1999D	The Fortified Forest	二维平面上给定 N 棵树，每棵树有一个价值和砍这棵树所能得到的篱笆长度，现在要求砍掉一些树使得所得篱笆能够将剩下的树围起来，要求剩下树的权值尽可能大，权值一样要求砍掉的树尽可能少。N<=15	因为 N 比较小，我们可以枚举砍掉哪些树，然后剩下没有砍掉的求凸包的周长。	时间复杂度：O(2^n*nlogn) 空间复杂度:O(n)
1999E	Trade on Verweggistan	给定 N 组物品，每组物品有 mi 个，如果想买第 i 个物品，必须买第 1 到第 i-1 个物品，购买每个物品有一个花费 cost 元，你会将你买到的所有物品以每个 10 元的价格卖掉，问如何买这些物品能使得你的获利最大。买几个物品可以得到最大获利，如果多解输出最小的 10 个解。N<=50 每组物品个数<=20	首先所有组之间互相独立，对于一个组只需要选择那个最优的就可以了，最后将每个组的最优值加起来就是最大获利。合并组与组之间的方案时，只需要保留最小的 10 个值，然后暴力合并。	时间复杂度：O(nm) 空间复杂度:O(nm)
2000B	According to Bartjens	给定一个长度<=9 的数字串，要求在里面添加+-*三个运算符号，要求添加完后的表达式满足第一个数字前和最后一个数字后没有运算符号，数字没有前导零，没有两个运算符号相邻，至少添加一个运算符，表达式的值为 2000。求所有满足条件的表达式按字典序输出.	暴力枚举所有可能的方案然后写一个表达式求值判断是否等于 2000。可以讲所有的可行解扔到一个 vector 里，最后 sort 一下就能满足字典序的要求了。	时间复杂度：O(4^n*n) 空间复杂度:O(n)
2000C	Cutting Chains	给定 N<=15 个链环，有些链环彼此相连，现在要求所有链环排成一条链，求必须打开的链环的数量。	N 比较小，我们暴力枚举打开哪些链环，然后判断是否可行。因为要接成一条链，所以没有打开的链环必须是一些单独的链环和一些链，而且单独的链环和链的数目不能超过断开链环的数目+1。	时间复杂度:O(2^n*n) 空间复杂度：O(n)
2000E	Internet Bandwidth	N<=100 个点 M<=N^2 条无向边求从 S 到 T 的最大流	最大流	时间复杂度:O(n^2m) 空间复杂度:O(n+m)

2000F	Page Hopping	N<=100 个点 M<=N^2 条有向边，求所有点对间最短路的平均值	Floyd 求最短路	时间复杂度:O(n^3) 空间复杂度:O(n^2)
2001A	Airport Configuration	机场有 N 个入口和 N 个出口，现在有 N 个城市，每个城市到达的航班对应一个入口，离开的航班对应一个出口，现在告诉你从城市 i 来想到城市 j 去的人有多少，他就需要从城市 i 对应的入口走到城市 j 对应的出口，入口 i 和出口 j 间的距离定义为 |i-j|+1，给 Q 个询问告诉你每个入口对应哪个城市每个出口对应哪个城市，求所有人需要走的距离总和是多少。将距离总和升序输出。N<=25 Q<=20	对于每一个询问 N^2 枚举从城市 i 来到城市 j 出去的游客需要走的距离乘上人数，累加起来就是每个询问的答案，最后排序输出。	时间复杂度:O(QN^2) 空间复杂度:O(N^2)
2001B	Say Cheese	三维空间有一些点，每移动 1 单位长度消耗 10 单位时间，一旦离点 i 距离小于 ri 那么移动将不花费时间，给定起点和终点问最短时间。	点 i 到点 j 耗费的时间为他们之间的欧几里得距离减去 ri 和 rj 和 0 去最大值然后乘 10。然后做一遍最短路即可。	时间复杂度:O(n^2) 空间复杂度:O(n)
2002A	Balloons in a Box	三维空间给一个盒子和一些点，按照任意顺序往每个点放置一个气球，气球将膨胀到碰到其他气球或者盒子边界停止，求如何放置使得盒子剩余空间最小。点数 N 小于等于 6	枚举气球的放置顺序，然后每个新放置气球和之前放置的气球和盒子边界判断半径。	时间复杂度:O(N!*N^2) 空间复杂度:O(N)
2002C	Crossing the Desert	二维平面上给定 N 个点，点对之间的距离是他们的欧几里得距离，现在你从一个点出发，有一个容量为 V 的背包，你每走一个单位的路程需要消耗一单位的水和一单位的食物，食物只能在出发点购买，水可以在任意点购买，但是任何时候身上携带的水和食物不能超过 V，食物可以存放在其他点然后再取走，问从起点到终点的最短路径。	令 f[i]表示从点 i 走到终点所需的最短路径，最后 f[1]即为所求。考虑逆推 f 数组，假设有边 i->j 那么由 f[j]推 f[i]，设两点间的距离为 len，第一种情况为 V>=2*len 这种情况下可以一次从 i 走到 j，第二种情况为 V>=3*len 这种情况下需要折返，否则 i 不能走到 j。由于转移过程存在后效性，我们用任意最短路解出 f 数组即可。	时间复杂度:O(N^2) 空间复杂度:O(N)
2002E	Island Hopping	二维平面上 N 个岛屿，每个岛有一个 m 个人，两个岛间的距离是欧几里得距离，现在想建一些电缆，把 N-1 个岛都与 1 号岛直接或间接连通起来，求最小的距离和是多少，在第一问的前提下，所有电缆同时开始建，求每个人到点 1 连通时间的平均值是多少。	第一问是裸的最小生成树，第二问就是任意生成一颗生成树然后计算出来每个点到 1 号点路径上的最大值，因为最小生成树的环切性质，所以所有最小生成树本质上是一样的。	时间复杂度:O(N^2) 空间复杂度:O(N)
2002H	Silly Sort	给定一个元素互不相同的序列，每次可以交换任意两个元素，代价是他们的和，求将序列升序排列所需的最小代价。N<=1000，元素值<=1000。	整个过程是若干个轮换，轮换间可以互相独立考虑，一个轮换的最小代价分为两种情况，第一种是拿轮换里的最小值不	时间复杂度:O(N) 空间复杂度:O(N)

			断交换，第二种是把所有数里的最小值换入轮换里，然后交换，两种情况取最优值就行，上述整个过程是线性的，唯一的瓶颈在于找到每个值对应的值，如果用快排是 O(NlogN) 用计数排序做到 O(N)。	
2003H	A Spy in the Metro	N<=50 个地铁站排成一行，地铁在第 i 和第 i+1 个地铁站间的运行时间为 ti，现在有 M1<=50 辆从 1 开到 N 的地铁和 M2<=50 辆从 N 开到 1 的地铁，他们的出发时间是已知的，你可以在每个车站等待，也可以乘车，但是要求在 T<=200 单位时间后的你在站台 N。现在你希望等车的时间尽可能的少，求出最少的等车时间。	设 f[i][j]表示 i 时刻在车站 j 的最小等待时间，为 INF 表示不可达。考虑三种转移，第一种是呆在原地这种转移没有任何限制条件但是耗时会增加 1 个时间单位，第二种是有从 1 开到 N 的车，那么可以向右边站台转移，时间也相应推移，第三种是有 N 开到 1 的车，向左边站台转移，最后 f[T][N]就是答案。	时间复杂度:O(NT) 空间复杂度:O(NT)
2003J	Toll	给定一张无向图，节点分为 AB 两类，A 类节点标小写字母，B 类节点标大写字母。给定起点和终点，要求从起点开始向终点运输货物，沿边进入 A 类节点需要缴纳 1 件货物，进入 B 类节点需要缴纳 $\lceil w/20 \rceil$ 件货物（W 为当前货物总量），现在需要运输 P 件货物到终点，求从起点开始最少需多少货物。边数≥0，1≤P≤1000。	定义 f[i]为从点 i 开始向终点运输货物所需的最小货物数，定义 W(x,y)表示进入节点 x 缴纳货物后至少剩余 y 件货物，则进入节点 x 前最少需要有几件货物，当 x 为 A 类节点时 W(x,y)=y+1，当 x 为 B 类节点时 W(x,y)=y+$\lceil y/19 \rceil$。f[i]=w(j,f[j])。注意到 w 始终为非负整数，我们选用最短路不断更新答案。	时间复杂度:O(N^2) 空间复杂度:O(N)
2004H	Tree-Lined Streets	二维平面上有 N<=100 条线段，在线段上种树，距线段间的交点 25 米范围内不能种树，同一条线段相邻两棵树的距离至少为 50，求最多的种树数目。	线段间互不干扰，对于一条线段求出所有交点，然后对于一个端点按照距离排序，一段距离为 x 的线段上可以种的树是 $\lfloor x/50 \rfloor$+1。如果线段左端是交点，x 就减 25，右端是交点 x 再减 25。	时间复杂度:O(N^2) 空间复杂度:O(N)
2005E	Lots of Sunlight	给定日出时间和日落时间，给定一些垂直于 x 轴且垂足在 x 轴上的线段，地平线是 x 轴，太阳匀速运动，询问某些点在哪些时候能看到太阳。线段数 N<=100，询问数 Q<=1000。	求出线段上端点与询问点连线和 y 轴的夹角，大于 90°的取一个最小值，小于 90°的去一个最小值，然后按比例算出起始和结束时刻。	时间复杂度:O(NQ) 空间复杂度:O(N)
2005G	Tiling the Plane	给定一个多边形，问无限复制下去能否不重不漏的覆盖无限大的平面，保证覆盖只有两种情况：在多边形边界上顺次存在四个点 A,B,C,D（不一定要是多边形的顶点），使得 A 到 B 的边界与 D 到 C 的边界重合，B 到 C 的边界与 A 到 D 的边界重合。在多	设 f[i][j]表示从第 i 个点开始逆时针和从第 j 个点开始顺时针能匹配的最远距离。然后枚举分解点{A，A+N/2}，分界点{B,B+N/2}，分界点{C,C+N/2}，用 f 数组判断匹配情况。	时间复杂度:O(N^3) 空间复杂度:O(N^2)

		边形边界上顺次存在六个点 A,B,C,D,E,F（不一定要是多边形的顶点），使得 A 到 B 的边界与 E 到 D 的边界重合，B 到 C 的边界与 F 到 E 的边界重合，C 到 D 的边界与 A 到 F 的边界重合。多边形边长<=50。		
2005H	The Great Wall Game	N*N 的棋盘上有 N 个棋子，每个棋子可以上下左右移动，求将所有棋子排成一行一列或者对角线所需的最小移动步数。	考虑排成一行，行和列分开考虑，因为排成一行，所以所有点在列 1 到列 n 都有分布，将 y 坐标排序，从小到大依次对应 1 到 n，再考虑放到哪一行上，将横坐标排序，中位数即为所求。列和行情况类似。对角线的情况，考虑做最优匹配，左边 N 个棋子，右边 N 个对角线的位置，然后两个对角线做两次最有匹配即可。	时间复杂度:O(N^3) 空间复杂度:O(N^2)
2005I	Workshops	给定 N<=1000 个会议，每个会议有 a 个人，需要 b 的时间，有 M<=1000 个会议室每个会议室容量 c 人，可用时间为 d，问如何安排会议可以使得未被安排的会议最少，满足上述条件下满足未被安排的人最少。	会议和会议室按照可用时间排序，依次对每个会议室找一个能放进来的会议里面人数最多的。	时间复杂度:O(Nlog N) 空间复杂度:O(N)
2005J	Zones	N<=20 个集合，告知每个集合的元素数目，告知一些同属于不同集合的集合的交的信息：属于哪些集合，元素数目是多少。现在要求在 N 个集合里面选 M 个集合，使得包含元素尽可能多。	枚举所有可能的情况，将所选的集合的元素数加起来，减掉他们之间的交集。	时间复杂度:O(C_N^M*N) 空间复杂度:O(N^2)
2006A	Low Cost Air Travel	N<=100 个城市 M<=20 条航线，航线依次飞过一些城市，票价为 Ci。你可以在起点上飞机并选择任意一个经过的城市结束航程。现在给 Q<=20 次询问，每次询问依次给出 C<=10 个城市，表示需要按顺序访问这些城市，但是不一定是连续的访问，问如何乘坐航班有最少的花费。	对于每个询问，将状态分层考虑 f[i][j]表示满足前 i 个询问的情况下位于城市 j 时的最小花费，现在可以选择任意一条航线做转移，转移只会向当前层或者更靠后的层转移，同层之间的转移可以用最短路，层间的转移用 DP。	时间复杂度:O(CN^3) 空间复杂度:O(NC)
2006B	Remember the A La Mode!	左边 N<=50 个饼片，右边 M<=50 个冰激凌，第 i 种饼片搭配第 j 种冰激凌能卖 aij 元，现在第 i 种饼片有 bi 个，第 j 种冰激凌有 ci 个，求在冰淇淋和饼片搭配最多的情况下，能卖的最少价格总和和最多价格总和	源点向饼片连接流量为饼片数，费用为 0 的边，饼片向冰激凌连接流量无穷费用为价格的边，冰激凌向汇点连流量为冰激凌数，费用为 0 的边，最小费用最大流即为最小值。边权反向再求一次最大值。	时间复杂度:O(N^4) 空间复杂度:O(N^2)
2006D	Bipartite Numbers	求一个比 x<=99999 大的最小二段数是 x 的正整数倍，二段数的定义是前一段 m 个数字都是 s，后一段 n 个数字都是 t，要求 m,n>1,s 不等于 t。输出 m、s、n、t。	首先答案不会超过 x+1，因为令 a1=1,ai+1=ai*10 在 mod x 的意义下循环节长度最大为 x。枚举 m+n 和 s，将之前长度<m+n 的形如	时间复杂度:O(100x) 空间复杂度:O(10x)

			111…111*i 的数 mod x 的值都存入数组内，枚举比 s 小的值 t 和比 s 大的值 t，这样可以判断出在 n+m 位数中是否存在一个值满足条件，如果存在的话，枚举 t 和 n。	
2006E	Bit Compressor	给定一种 01 串压缩规则，对于连续的 x 个 1，且左侧不为 1 右侧也不为 1，将它替换为 x 的二进制之后长度变小了就替换，现在给定一个压缩后的串问长度为 L 且串中 1 的个数为 N 的原串是否唯一。	从第一位开始，每碰到一个左侧不为 0 右侧不为 1 的就考虑替换，向右侧找只要能找到满足条件的就尝试替换，连续的两个 1 或者单独的一个 1 可能不需要替换。当搜到的答案大于 1 或者长度大于 L 或者 1 的个数大于 N 的时候剪枝。	时间复杂度：未知 空间复杂度：O(L)
2006G	Pilgrimage	一个队伍里有若干个人和若干公共财产，现在有 4 种操作，IN K 表示 K 个人加入了，加入之前这时所有人恰好可以平分现在的公共财产，K 个人缴纳他们的那份公共财产，OUT K 有 K 个人离开了，离开之前所有人恰好可以平分现在的公共财产，K 个人带走自己的那份财产，COLLECT K 每个人上交 K 元钱给公共财产，PAY K 公共财产支出 K。现在告诉你这些操作问你可能的总人数有哪些，或者>=某个人数都可行。K<=2000，操作数 N<=50。要求队伍中自始至终至少有 1 个人。	可以计算出每个时刻相对最开始减少了多少人，取最大值加 1 是答案的下界。因为初始的财产不确定，而 collect 操作又不会对平分产生影响所以忽略所有 collect 操作，在第一个人数变化操作之前和最后一个人数变化操作之后的所有 pay 操作都是没有意义的，我们可以通过调整初始财产达到要求。相邻的 pay 操作可以合并成一个。现在如果没有有效的 pay 操作了那么>=人数下界就都可行了。因为 pay 操作后一定跟着一个人数操作所以 pay 完之后财产仍为人数的整数倍，也就是说任何时刻当前人数都应是当前 pay 的一个约数，我们枚举 pay 的约数然后取交集即可。	时间复杂度:O(N+sqrt(NK)) 空间复杂度:O(N+sqrt(NK))
2006I	Degrees of Separation	N<=50 个点 M 条边无向图，求是否连通，是的话求最长的点对间最短路。	Floyd 求多源最短路。	时间复杂度:O(N^3) 空间复杂度:O(N^2)
2006J	Routing	N<=100 个点 M 条边的无向图，求点数最小的子图使得点 1 和点 2 可以互达。	第一步 floyd 求出任意两点间最少经过几个点 d[i][j]。设 f[i][j]表示从 1->i 和从 j->1 两条路径上最少经过多少个点，重复的点算一次。设下一个点为 k 考虑两种转移，从 i 到 k f[k][j]=f[i][j]+d[i][k]和从 k 到 j 转移类似。再考虑 i->j 路径重复使用的情况 f[j][i]=f[i][j]+d[i][	时间复杂度:O(N^4) 空间复杂度:O(N^2)

			j]-1,因为 i 和 j 都重复经过所以需要-1。用最短路维护转移，最后 f[2][2]就是答案。	
2007A	Consanguine Calculations	给定父母的血型求孩子的血型或者给定父母其中一人和孩子的血型，求另一人的血型。血型包含 ABO 型和 RH 型。	如果是已知父母的血型，就枚举父亲的等位基因和母亲的等位基因然后组合。如果是求父母一方的，可以枚举答案，然后 check 是否能得到孩子的血型。	时间复杂度:O(1) 空间复杂度:O(1)
2007G	Network	有 N<=5 条信息，一共被拆成了 M<=1000 块，顺序被打乱了。现在 M 块信息依次输入，你可以将输入的信息寄存在缓冲区也可以直接输出，要求每条信息必须是连续按顺序输出来的，信息间顺序不计，同一条信息不一定是同一时刻输出。求缓冲区的最小大小。	枚举输出信息的顺序，然后模拟整个过程，输入一块信息，如果当前输出的信息是这一条，并且这条信息在这块之前的都输出了就输出这块信息，否则就放到寄存器内，当一条信息输出完之后开始输出下条信息。	时间复杂度:O(N!M) 空间复杂度:O(M)
2007I	Water Tanks	给 N 个水箱从左到右排成一行，第 i 个水箱的高度是 hi，水箱的底面积都是 1 平方米，第 i 个和第 i+1 个水箱通过一个管子相连，高度是 pi，pi<pi+1,除了第一个水箱的顶端开口外，其余的水箱都是封闭的。现在问你从第一个水箱开始缓慢注水，问注入水的体积是多少。	从第 1 个水箱开始往后考虑 3 种状态，第一种是当前水箱注水不到 pi 就停止了，第二种是不到 pi+1 就停止了，第三种是超过 pi+1 了，并且能继续向之后的水箱注水。分类讨论三种情况，求出高度 h，计算内部气体压强的变化，用底面压强相等列出方程，在注满水是，第一个水箱的底面压强恒定。	时间复杂度:O(N) 空间复杂度:O(N)
2008A	Air Conditioning Machinery	给一个长方体 X*Y*Z,给一个入口和入口方向，给一个出口和出口方向，给 6 个一模一样的管道，可以任意方式放置，但是两个连接口是一定的，要求放置尽可能少的管道使得从入口入从出口处，管道不能超出给定长方体的范围。	设定状态 f(x,y,z,dir)表示位于(x,y,z)以及现在的朝向，然后枚举管道的放置方式，判断是否交叉和是否超出范围，最后看达到终点的最少步数。搜索深度要求小于等于 6。	时间复杂度：未知 空间复杂度：O(V)
2008B	Always an Integer	给定自变量为 n 的多项式,问当 n 取任意正整数时这个多项式的值是否恒为整数？	对于 n 次有理系数多项式 f(x),若存在 n+1 个连续整 p 使 f (x)的值都是整数值，则 f(x)为整值多项式。我们将字符串解析之后，依次从 1 到 n+1 带入即可。	时间复杂度:O(n^2) 空间复杂度:O(n)
2008E	Huffman Codes	给定N个字符对应的haffman编码，要求出现次数小的作为作为左孩子，求 N 个字符的所有可行频率分布 x%，要求 x 为正整数，不同顺序的算一种。	将给定的信息对称一下，得到的是出现次数大的作为左孩子。这样满足同一层里面的从左到右单调不降，当前层的最左端值小于等于上一层的最右端值。一层一层从左到右搜索，可以直接用位运算来维护，要求当前两个孩子的频率和等于	时间复杂度：未知 空间复杂度:O(N)

			父亲且当前的频率值小于等于上一个频率值。	
2008F	Glenbow Museum	直角多边形的角序列定义为从一个点出发逆时针沿边行走，如果左转就是 L 右转就是 O 最后这个 LO 序列就是角序列。给定角序列长度 N，求满足条件的角序列个数。满足要求的角序列对应的其中一种多边形可以在其中一个点看到多边形边上的所有点。	首先 N 个字符 L 比 O 多 4 个，恰好能绕一圈回到原点，也就是说如果 L 不是偶数就无解。其次两个 O 不能相邻，问题就变成了求一个长度为 N 的序列 0 比 1 多 4 个且 00 不相邻（一首一尾也不行）的数量。枚举两端都不是 O，左端是 O 和右端是 O 三种情况，设所需的 L 的数目为 a 个，所需 O 的数目为 a-4 个，答案为 C(a-4,a-1) +C(a-5,a-1)*2	时间复杂度:O(1) 空间复杂度:O(1)
2008I	Password Suspects	给定 N 个模式串求有多少个长度为 M 的字符串包含所有 N 个字符串，如果方案数≤42，按照字典序打印所有字符串，字符串中所有字符皆为小写英文字母'a'-'z'。1≤N≤10，1≤M≤25。	定义 f[i][j][S]表示长度为 i 的串在自动机上匹配到了 j 节点,包含的模式串集合为 S 时的方案数，定义 next[i][j]表示在自动机的 i 节点后添加字符 j 到达了哪一个节点。枚举添加字符 c，则 f[i][j][S] 可以向 f[i+1][next[j][c]][S|bit[next[j][c]]]转移。当方案数≤42 时，需要输出方案。设 g[i][j][S]表示在当前状态下往后添加字符能否构成一个满足条件的字符串。只有当 f[i][j][S]>0 且状态 (i,j,S) 的某个后继 (i',j',S') 满足 g[i'][j'][S']=true 时，g[i][j][S]=true。因为串的长度不超过 25，可行的串不超过 42 个，所以有用的状态数和可行的转移数为 25*42，将可行的状态离散化并用一个 vector 记下来所有的转移即可输出所有方案。	时间复杂度:O(2^N*MK) 空间复杂度:O(2^N*MK)
2008J	The Sky is the Limit	给定 N 个底边在 X 上的等腰三角形，求三角形并的轮廓线（不包括 x 轴和底边）的长度。	三角形两两求交点，扫出来所有的事件点 x，对于事件点求出来最大的纵坐标 y 使得 (x,y)在三角形上，相邻的两个事件点间，求两个坐标间的距离，将距离累加就是轮廓线长度。	时间复杂度:O(N^3) 空间复杂度:O(N^2)
2008K	Steam Roller	给定一张带权网格图 (N<=100*M<=100)，从起点加速或	将一个点拆成 8 个点 dir 表示从 4 个方向哪个方向来，inc	时间复杂度:O(NMlo

		者到终点减速或者拐弯前后的减速和加速，经过边的耗时多一倍，求从起点到终点的最少耗时。	表示耗时是否需要*2。用dijkstra+heap 算起点到终点的最短路。	gNM) 空间复杂度:O(NM)
2009A	A Careful Approach	N<=8 架飞机，每个飞机有一个降落区间[li,ri]表示可以在 li 到 ri 这段时间内降落，现在要你安排飞机降落的顺序和时间，使得相邻两架降落的飞机的最小间隔最大。	因为求最小值最大，我们可以用二分的方法二分这个最小值，转化为判定性问题。枚举飞机的降落顺序，依次判断可行性。	时间复杂度：O(N*N!log ANS) 空间复杂度:O(N)
2009B	My Bad	给一个逻辑电路，有 N 个输入，G 个门,U 个输出。门有四类与门 或门 异或门 非门，现在给定 Q 个输入和输出，让你判断哪个门坏了。有三种故障情况：总是输出 0 总是输出 1 总是输出相反的那个结果。	依次枚举哪个门坏了，是哪种故障，然后对于所有输入和输出判断一下看看是否都符合情况。	时间复杂度:(GQ(N+G+U)) 空间复杂度:((N+G+U)^2)
2009D	Conduit Packing	给定 4 个圆的直径，问包含这 4 个圆的最小圆的直径。	二分答案，下界是最大的 2 个圆的直径和，上界是 4 个圆的直径和，验证的时候可以看成是将 4 个小圆顺次放在大圆的圆周上，枚举所有可能的放置顺序，然后判断旋转角加起来和 2π的关系即可。	时间复杂度:(logR*4!) 空间复杂度:O(1)
2009E	Fare and Balanced	给一张有向无环带权图，要求给一些边加权，使得所有从点 1 到点 n 的路径的长度都相等，且同一条路径上至多只能改变一条边的边权。要求改变后从点 1 到点 N 的路径长度最小。点数<=5W，边数<=5W	如果有解的话，最终点 1 到点 N 的路径长度一定就是初始图里面点 1 到点 N 的最长路。求出从点 1 出发到所有点的最长路 f[i]以及点 1 出发到所有点的路径长度是否都相等 g[i]从点 n 倒着再求一遍 f_opp[i]和 g_opp[i]如果无解当且仅当存在任何一个点 g[i]和 g_opp[i]都是 false。一条点 i 到点 j 的边被改变当且仅当 f[i]+f_opp[j]+len<f[n]并且 g[i]=true 并且 g_opp[j]=true 并且 g[j]=false，此时 i 到 j 的改变值为 f[n]-f[i]-f_opp[j]-len。	时间复杂度:O(N+M) 空间复杂度:O(N+M)
2009F	Deer-Proof Fence	二维平面上有 N<=9 个点，现在你需要用一些篱笆将他们围起来，可以将他们围成若干部分，但是要求篱笆距离他们的距离最少是 R，求所需篱笆的最短长度。	定义 f[mask]表示将 mask 这个二进制数所表示的点都围起来所需的最小篱笆长度。f[mask]的初值就是将这些点围成一块所需的最小长度。这个长度就是这些点的凸包周长加上一个半径为 R 的圆的周长。f[i]=max(f[j]+f[i^j])	时间复杂度:O(3^N) 空间复杂度:O(2^N)

			(j 是 i 的子集)	
2009H	The Ministers ' Major Mess	N<=100 个议案，M<=500 名大臣。每名大臣最多只会向 4 个议案投票。投的票一定是同意或者反对。一个大臣感到满意需要至少有大于一半的建议被满足。你的程序需要指出是否每个大臣都能被满足，如果是，请指出哪些议案只能被通过或否决。如果这个议案一定要被通过输出 Y 一定不被通过输出 N 两种都可以输出?不能满足所有大臣输出 IMPOSSIBLE	注意到超过一半这个条件，也就是说一个大臣最多只有 1 个议案的结果与他投的票不同。我们考虑用 2-sat 来维护。如果大臣投的票数<=2 那么每个议案都必须被满足。如果票数>2 那么将可以有 1 个议案不被满足，这个不被满足的议案向其余三个投票的对立面连边。然后枚举每个议案是 Y 还是 N 逐个判断就能得到答案。	时间复杂度:O((N+M)^2) 空间复杂度:O(N+M)
2009I	Struts and Springs	给 N<=500 个窗户，窗户互不重叠包含关系呈树形结构，除了最外层窗户外，每个窗户横向有 3 个弹簧/支杆，分别连接左侧和外部，左右两侧，右侧和外部，竖向有 3 个弹簧/支杆，分别连接上侧和外部，上下两侧，下侧和外部，如果 3 个都是支杆，上/右侧支杆变成弹簧。现在给你每个窗户的大小和相对于最外层窗户的位置，给 Q<=500 个询问，每个询问将改变最外层窗户的大小，求每个窗户的位置及大小。	先排序确定每个窗户外面的窗户是谁。然后对于每一个询问从最外层窗户开始向内模拟，算出每个弹簧增长或缩短的比例，一个比较好的方法是只记录当前窗户相对于上一个窗户的坐标。	时间复杂度:O(NlogN+NQ) 空间复杂度:O(N)
2009J	Subway Timing	给一棵 N 个点的树，通过每条边有一个时间单位是秒，现在要将它向上或向下取整到整分钟，求如何改变边使得误差最大的两个点对误差最小。	二分最小的最大误差 mid。设 f[i][j]表示在点 i 子树中所有点到点 i 的最大正误差为 j 时最小的负误差是多少。如果要合并两颗子树，设第一个的误差范围是[-f[i1][j1],j1]第二个的误差范围是[-f[i2][j2],j2]那么当且仅当 j1+j2<=mid 并且 f[i1][j1]+f[i2][j2]<=mid 的是时候才可以。注意到答案一定不会超过 118，因为我们总可以将每个点相对于点 1 的误差控制在[-59,59]。	时间复杂度:O(N^3 logN) 空间复杂度:O(N^2)
2009K	Suffix-Replacement Grammars	给定若干个后缀替换规则（A,B）串 A 长度等于串 B,如果当前串 A 是某个字符串的后缀，那么可以将它替换成串 B。给定起始串和目标串以及若干替换规则，求最少步数。规则数 N<=100，字符串长度 L<=20。	在长度为 i 的后缀中，有用的后缀一共只有 2N+2 个。对于每个长度我们都做一次 floyd 这样就可以知道长度为 i 的后缀中的相互转移情况了，然后由长度为 i 的最短路数组可以推出来长度为 i+1 的最短路数组，这样整个复杂度是 LN^3，而我们将所有后缀按照长度排序，然后 floyd 的最外层按照	时间复杂度:O(N^3) 空间复杂度:O(N)

			从小到大的顺序枚举，最短路数组就有了继承关系，我们只需要做一次 floyd 就可以得到最后的答案。	
2010B	Barcodes	给定一种编码方式，对于某个编码的扫描结果输出是否合法，合法的话输出原串是什么。	首先要确定每个区域是属于宽区域还是窄区域，或者编码不合法。我们可以枚举区域的合法值，然后依次判断每个区域的宽度是在宽区域的误差范围内还是在窄区域的误差范围内。如果能将每个区域都分成宽窄两种区域，就将其转化为 01 串，然后将其每一位转码，注意 begin 和 end，正着和倒着各做一遍按题目要求输出即可。	时间复杂度：O(N) 空间复杂度：O(N)
2010C	Tracking Bio-bots	给定一个 N*M 的网格图，有 W 块矩形墙壁，从(0,0)出发可以向右或者向上走，问有多少个点出发不能到达（N-1,M-1)。N,M<=10^6 w<=1000	因为 N 和 M 很大，我们考虑只将墙壁所在的 x 和 y 坐标离散出来，一个(x1,y1,x2,y2)矩阵，X 轴上有用的事件点有 x1 和 x2+1，Y 轴上有用的事件点是 y1 和 y1+1,这样我们可以组成一个 2w*2w 的新矩阵，然后逐格递推可以得到不可达的离散化后的格子有哪些，计算出来对应原图中的面积。	时间复杂度:O(w^2) 空间复杂度:O(w^2)
2010D	Castles	N<=100 个城堡构成一个树结构，每个城堡有成功夺取城堡需要的最少士兵数量，在进攻中会死去的士兵数量和必须留守的士兵数量，你可以从任意一点开始发起进攻，然后军队必须沿树边移动，一条树边的同一方向只能经过一次。问最少需要多少人才能攻占整个城堡。	首先留守的和死去的士兵数量，可以合并为同一个数量，表示为攻打该城堡后不能带走的士兵数量。设 f[i][0]表示攻打 i 这个子树需要多少士兵，f[i][1]表示不能打走的士兵有多少。对于 i 的孩子，我们按照 f[i][0]-f[i][1]降序的顺序依次访问。需要新派的部队数是当前孩子所需军队数减掉之前剩余的军队数，消耗的军队数单纯相加即可。	时间复杂度:O(n^2 logn) 空间复杂度:O(n)
2010F	Contour Mapping	给一个边长是 d<=10 的正三角形构成的网格图，网格图有 s<=100 行，每行 p<=100 或 p+1 个三角形，三角形的顶点有一个高度，现在要求绘制整个地图的高度是 h<=1000 整数倍的等高线。同一高度区域内部等高线不画出来。	首先三角形内部的等高线一定会被画出来，我们分三种情况分类讨论一个三角形内部的等高线情况。(a,a,b)(a,b,b)(a,b,c)(a<b<c)我们用等差数列结合三角函数即可算出。前两种情况下等高线两端点都分布在(a,b)上，后面一种情况等高线有两种，第一种分布在	时间复杂度:O(sp) 空间复杂度:O(sp)

			(a,b)(a,c)上，第二种分布在(b,c)(a,c)上。然后我们考虑正三角形的边作为等高线的情况，一个正三角形边作为等高线，需要包含它的两个三角形的另外两个端点高度不相等，或者正三角形边是整个网络的边界。	
2010G	The Islands	二维平面上给定 N<=100 个点，从西向东按照 x 坐标严格递增排列。给定两个特殊点 P1 和 P2，要求找到最短的一条路径从点 1 出发到 N 经过 P1 或 P2 中的任意一个然后从 N 返回 1 经过 P1 和 P2 中剩余的一个，要求所有点除了 1 和 N 恰好经过一次，P1P2 分两次经过，求满足上述条件的最短路径。	我们可以看成是从点 1 出发的两条路径第一条必须经过 P1 第二条必须经过 P2 到达点 N，且中间不经过重复点。设 f[i][j]表示第一条路径到达 i 了第二条路径到达 j 了的最小路径，注意转移的时候要保证 max(i,j)前的点均被访问。 如果 i 不是 p2 那么 i 可以由点 1 访问 f[i][j]=f[i-1][j]+dis[i-1,i] f[i][i-1]=f[j][i-1]+dis[j,i] 如果 i 不是 p1 那么 i 可以由点 2 访问 f[i-1][i]=f[i-1][j]+dis[j,i] f[j][i]=f[j][i-1]+dis[i-1,i]。 另开两个数组记录 DP 路径输出方案。	时间复杂度:O(N^2) 空间复杂度:O(N^2)
2010I	Robots on Ice	给定一张 N<=8*M<=8 的 4 连通地图，要求从点(0,0)出发经过每个点恰好一次回到(0,1)，且要求在 nm/4 nm*2/4 nm*3/4 步的时候恰好经过(x1,y1)(x2,y2)(x3,y3)三个点。求可行的路线数。	从起点开始搜索，如果当前点到下一个点的目标距离大于可行步数的时候剪枝。如果没有在规定的时间就走到了(xi,yi)剪枝。如果只能向上下两个方向或者左右两个方向走，即绕进死胡同时剪枝。	时间复杂度:未知 空间复杂度:O(NM)
2010J	Sharing Chocolate	给一块 N*M 的巧克力,可以横着切或者竖着切成 a*M 和(N-a)*M 或者 N*b 和 N*(M-b)的小块，1<a<N 1<b<M 且 a 和 b 为整数，然后对小块重复切割。问能不能把巧克力切成 T 块，每块恰好是 Ai 大小。	area[mask]表示 mask 这个二进制数所表示的小块面积和是多少。f[mask][i]表示能否将一块 i*area[mask]/i 大小的巧克力能否分割成 mask 所代表的的小块。其中 i*i<=area[mask]最后 f[2^T-1][min(N,M)]就是答案。	时间复杂度:O(3^T max(N,M)) 空间复杂度:O(2^T max(N,M))
2011A	To Add or to	A 操作表示+a，M 操作表示*m，给定区间[p,q]要求输出一个长度最短	首先当 m=1 的时候肯定不需要乘法操作，而当 m>1 时最多进	时间复杂度:O(log^

	Multiply	的由 A 和 M 组成操作序列，使得 [p,q]内的所有数操作后落在区间 [r,s]内。所有数据范围<=10^9	行约 30 次乘法就超出了 10^9 的范围了，所以我们可以暴力枚举进行了多少次乘法操作。注意到对于[p,q]这个区间加法导致平移，乘法导致平移+扩大，我们可以求出若干次乘法后区间的长度，这样可以得到一个加的数的范围[l,r]。然后对 l 在 m 进制下分解成 pi*m^i 这样的形式相加，然后在不超过 l 的情况下从低位开始逐一做进位考虑。	2N) 空间复杂度:O(logN)
2011C	Ancient Messages	要求你在一个 NM 的 01 矩阵中识别出来 6 种图像。N,M<=200	不同图像的"洞"的数目不一样，写个 floodfill 数一下就可以了。	时间复杂度:O(NM) 空间复杂度:O(NM)
2011E	Coffee Central	给定一个 N*M 的网格图和 T 个咖啡厅，现在给 Q 个询问，每个询问要求选择一个点(x,y)使得到这个点曼哈顿距离<=m 的咖啡厅的数目最多，输出最南最西的一个。	将坐标旋转为(x-y,x+y)然后枚举范围内的每个点，然后用前缀和 O(1)距每个点 m 内的点的数目。	时间复杂度:O(QNM) 空间复杂度:O(NM)
2011F	MachineWorks	有 N 件物品，可以在第 di 天买进第 i 件物品，买进需要花费 pi 元，每天可以带来 gi 的利润（除了买进和卖出的一天)，可以在买后任意一天以 ri 元卖掉。第 0 天你有 C 元钱，问到第 M+1 天你的最大收入。	f[i]=max(f[j]-p[j]+r[j]+g[j]*(d[i]-d[j]-1)){j<i&&f[j]>=p[j]} 令 yj=f[j]-p[j]+r[j]-g[j]*(d[j]+1) xj=g[j];ki=-d[i]x 和 y 单调，用 CDQ 分治维护一个凸壳，斜率是单调递增的。	时间复杂度:O(NlogN) 空间复杂度:O(N)
2011G	magicsticks	给定一排 N 条线段，可以将相邻的一些线段拼成一个多边形，多边形上边的顺序和序列中的顺序一致，问能拼成的多边形的总面积最大是多少。	f[i][j]表示从 i 到 j 能拼成的最大面积。 f[i][j]=max(f[i][k],f[k+1][j])求 i 到 j 的线段拼成一个多边形的面积，可以用二分来求，这个多边形的顶点一定在同一个圆上，我们二分这个圆的半径，分成圆心在多边形内外两种情况讨论，通过计算边绕圆一周的角度来二分。注意到如果 i 到 j 没有拼成一个大的多边形，一定是将其中最大的一条边删掉了，于是我们可以将转移简化为 f(i,j)=max(polygon(i,j),f(i,k-1)+f(k+1,i)) k 是 i 到 j 中最大边的位置。	时间复杂度:O(N^2logN) 空间复杂度:O(N^2)
2011H	Mining Your Own	N<=5w 个点 M 条边，现在想在一些点上修建向外的通道，使得删任意一	做点双连通分量，如果一个连通分量只连接一个割点时答案	时间复杂度:O(N+M)

	Business	个点，其余点都能通向外面。求最少修建通道数和方案数。	+1（割点断裂的话就独立了），方案数*连通分量大小（不含割点）。如果没有割点答案=2（如果只修建一个通道需要考虑通道自身塌陷），方案是N*(N-1)/2	空间复杂度:O(N+M)
2011I	Mummy Madness	你在原点，现在有 N 个木乃伊，你和木乃伊每秒都可以向周围 8 个方向移动一步，问你最多能走多少步才被木乃伊抓住。	首先二分答案 mid，你可以达到的点是(-mid,-mid,mid,mid)这样一个大的正方形，然后用其余的僵尸可达的正方形和你可达的做交，然后在这个正方形内部做扫描线，看是否存在一个点没有被覆盖过。用线段树维护区间加减区间最小值即可。	时间复杂度:O(Nlog^2N) 空间复杂度:O(N)
2011J	Pyramids	给定三种金字塔，第一种是V=i^2+(i-1)^2+…+1^2 第二种是 V=i^2+(i-2)^2+…+1^2 第三种是 V=i^2+(i-2)^2+…+2^2 现在给定 Q 个询问，每个寻味一个数 N<=100W 求用尽量少的上述三种金字塔拼出数 N 的，要求不能用重复的金字塔。	先求出 V<=100W 以内的上述三种金字塔，三种金字塔种类数在 200 左右，然后做一个 0/1 背包求出 f[i]表示拼出数字 i 的所需的最小金字塔数。对于给定的数字 N 搜索出一个可行的方案，由于限定了步数所以搜索速度非常快。	时间复杂度:O(kN+Qsearch(ans)) 空间复杂度:O(N)
2011K	Trash Removal	给一个 N 条边的多边形垃圾，想将这个多边形垃圾投放到一个管道中，求这个管道最小宽度。	求这个多边形的凸包，然后枚举每条边三分离他最远的点的距离是多少取最小值。	时间复杂度:O(NlogN) 空间复杂度:O(N)
2012A	Asteroid Rangers	三维空间给你 N<=50 个点的坐标和他们各自的移动的速度，问最小生成树改变过几次。	枚举所有边 A 和边 B，记录当 A>B 时的时间点，然后将时间点排序，扫描到当前时间点时，如果边 A 在最小生成树中，就暴力重建生成树并判断与原来的树是否相同。时间复杂度看上去是 N^6 实际上有用的决策点非常少。	时间复杂度:O(ans*N^2+N^4) 空间复杂度:O(N^4)
2012B	Curvy Little Bottles	给定一个 N<=10 次多项式曲线和 xlow 及 xhigh，求 xlow 到 xhigh 间的曲线绕 x 轴旋转一周所构成的柱体体积是多少，给定定值增量 inc，每隔 inc 体积做一次标记，只标记 xlow 开始(不含 xlow)的前 8 个标记，问每个标记距离 xlow 的距离是多少。	曲线 y=f(x)，直线 x=a，x=b 及 x 轴围成图形，绕 x 轴旋转一周所构成的旋转体体积Vab为：$V_{ab}=\int_a^b f^2(x)dx$ 二分每个标记的位置。	时间复杂度:O(N^2+logM) 空间复杂度:O(N)
2012C	Bus Tour	N<=20 个点 M 条边带权无向图，要求求一条从 1 到 N 再回到 1 的路径，使得去的时候访问 1 到 N 的所有点回来的时候访问 N 到 1 的所有点，且去	先求出每两个点间的最短距离。将整个过程分成四部分从 1 开始访问前 h/2 个旅馆，访问剩下的旅馆到 N，从 N 开始	时间复杂度:O(2^N*N^2) 空间复杂

		的时候的访问前 h/2 个旅馆和回来的前 h/2 个旅馆一样(顺序可以不同)。h=N-2，点 2 到点 N-1 是旅馆。经过不一定访问。	访问前 h/2 个旅馆，访问剩下的旅馆到 1。我们做 4 次集合动态规划，然后枚举访问的前 h/2 个旅馆是哪些，将 4 段路程组合起来取一个最小值。	度:O(2^N*N)
2012D	Fibonacci Words	F[0]=0 F[1]=1 F[2]=10 F[3]=101 … F[n]=F[n-1]+F[n-2] 给定 N 和模式串 P 求 P 在 F[N]里出现的次数。N<=100 length(P)<=10w	找出长度>=length(P)的第一个 i。令 G[0]=F[i] G[1]=F[i+1] G[n]=G[n-1]+G[n-2] 我们需要考虑的就是 P 在 G[0]里面的出现次数，P 在 G[1]里面的出现次数，P 在 G[0]+G[1],G[1]+G[1],G[1]+G[0]里面的出现次数，然后乘上各自的系数相加就是出现的次数。	时间复杂度:O(length(P)) 空间复杂度:O(length(P))
2012E	infiltration	N 个单位，任意两个单位要么 a 控制 b 要么 b 控制 a，现在给你控制的情况，要求你选择最少的点使得所有点都被控制，被选的点也属于被控制。	答案很小，使用迭代加深搜索来搜索答案，考虑用 bitset 来优化整个过程。	时间复杂度:O(N^ans) 空间复杂度:O(N^2)
2012F	Keys	大写字母代表钥匙“A”-“M”表示第一种钥匙其余的是另一种钥匙，小写字母代表环，钥匙都在环上，环之间相互连接构成森林。现在你可以取下钥匙或者取下环，接上钥匙或者接上环，要求最小化对钥匙的操作再最小化对环的操作，使得同一种钥匙都在一个连通块里。	因为首先最小化对钥匙的操作，所以如果一个环上只有一种钥匙可以先不用动，也就是说我们枚举那些环上有两种钥匙的保留哪种钥匙在环上，这样的环不会超过 13 种，然后我们枚举将取下来的两种钥匙各放在哪个环上。现在问题变成了每种环有一个属性 0/1/2，现在要求最少的操作把为 1 的都放在一个连通块内，为 2 的都放在另一个连通块内，为 0 的无所谓。我们可以用树形 DP O(N)的解决这个问题，设 f[i][0/1/2]表示当前与父亲相连的这个连通块的种类。是 0 的不需要断开，如果父亲是 1 而孩子是 2 需要断开然后将孩子里的 2 合并在一起。父亲是 2 和孩子是 1 同理。如果父亲和孩子一样那么不用动，仍然保持连接。最后 f[root][0/1/2]取最优值就是答案。	时间复杂度:(2^(N/2)*N^3) 空间复杂度:O(N^2)
2012G	Minimum Cost Flow	三维空间里有 N<=400 个点，每个点上有一些空洞，补一个空洞需要花费 0.5，将两个不同点上的空洞连接起来需要花费 dist(i,j)，这样两个	首先枚举施加的水压到达的最高高度，然后将比这个高度低的点选出来，将原来已经相连的合并成一个连通块，空洞数	时间复杂度:O(N^3) 空间复杂度:O(N^2)

		点就连通了，有一些点相连了，给一个起点和一个终点，你可以在起点处加任意多的水和水压，要使得 T 充满水且整个系统不漏水，求最少费用。	也合并起来，然后从起点开始运行任意最短路算法即可。注意起点和终点所在连通块只需要有一个空洞，而路径上其他的连通块至少需要两个空洞。且不会一个连通块不会在最短路径中重复出现。	
2012K	Stacking Plates	有 N 堆盘子，每堆盘子有 hi 个，从上到下直径保证不降。现在可以将一些牌子从一堆上拿下来新成立一堆。或者是合并两堆盘子，要求下面那堆的最上面那个盘子大于等于上面那堆的最下面的盘子。求最小操作次数。	假设最后的拼成的一大堆盘子，按照原来所属的堆一共能分成 k 块，那么答案就是 2*k-n-1。现在就是要使得 k 尽可能的小。将盘子按照大小排序，令 f[i][j]表示前 i 种大小的盘子最后一个是来自第 j 堆的现在一共有多少块。考虑第 i+1 种大小的盘子分别来自第几堆。然后枚举一下放在最后一个的哪个，然后前面判断一下有没有和 j 一样的，如果放在最后一个的是 j 需要特判，如果都来自同一堆也需要特判。现在的复杂度是 mn^2，我们将 f[i][j]用前缀和后缀维护一下可以做到 mn	时间复杂度:O(NM) 空间复杂度:O(NM)
2012L	Takeover Wars	两个人各有一组数，分别有 N 个和 M 个，每个人每次可以把自己的两个数合并起来变成一个更大的数，或者选择现在的一个数和对面一个比自己小的数，然后删掉对面的数，谁手中先没有数，谁就输了。问谁能赢。	设当前操作者是 A，另一个人是 B，如果 A 的最大值小于等于 B 的最大值，A 一定选择合并。否则，如果 A 只有一个数了，A 选择删掉 B 的最大值。如果 A 手中的数多于一个，判断 A 的次大值和 B 的最大值的大小，如果 A 的次大值大，那么合并最大和次大值，否则比较 A 的最大和次大值之和与 B 的最大和次大值之和的大小，如果 A 的较大就合并最大次大，否则就删除 B 的最大。	时间复杂度:O(NlogN+MlogM) 空间复杂度:O(N+M)
2013A	Self-Assembly	给定 N<=4W 种方格，方格的四个边上有”X±”或者”00”的标记，只有当第一个字母相同第二个一个加号一个减号时两个边才能匹配，”00”不能匹配任何边。求能否构造出一个无限大的图形。允许任意旋转翻转变换且方格数目无穷。	将每种边都建立一个节点，考虑同一种方格上的两条边，由 i 向 j’连边。然后从每个节点出发看看是否存在自环，如果有自环就存在无限大的图形。	时间复杂度:O(N) 空间复杂度:O(N)
2013B	Hey, Better Bettor	一种赌博游戏，每次花费 1 元，赢了得到 2 元，输了不获得钱。每次你赢的概率为 P<50%，当你结束赌博时，如果你出于亏损状态，赌场会返还你亏损数额的 X<100%。问在最优策略	令 f(i)表示收益 i 元时的期望总收益。 f(i)=max(i,i*(1-x),p*f(i+1)+(1-p)*f(i-1)); 容易发现当我们赢得 a 元或者	时间复杂度:O(log^3ans) 空间复杂度:O(1)

		下的期望收益。	输了 b 元是就停止赌博。f(a)=a,f(b)=b,f(n)=p*f(n+1)+(1-p)*f(n-1)，已知数列的两项，我们可以通过特征根解出这个数列的通项。接下来就是如何确定 a 和 b 的值，我们发现固定 a 时 f(0)关于 b 是单峰函数，b 取最优时，f(0)关于 a 是单峰函数。我们三分套三分解出来 a 和 b 即可。	
2013C	Surely You Congest	N<=2500 个点 M<=50000 条边带权无向图，有 C<=1000 辆车，每辆车都会沿最短路径向点 1 开，但是同一时刻每条道路的同一位置上最多只有一辆车通行，问最多有多少辆车能到达点 1。	首先构造出来关于点 1 的最短路图，很明显这些车只会在最短路图上开，否则一定不是最短路径。最短路图是一张有向无环图，我们考虑两个点 i 和 j 如果点 i 和 j 到达点 1 的距离不同，他们不可能在同一时刻位于同一条路的同一位置。所以我们考虑将到达点 1 距离相同的一组点拿出来，然后原点向他们连流量为 1 的边，最短路图里面的边也连流量为 1 的边，求原点到点 1 的最大流就是这一组车的最大值，将所有组的最大值加起来就是所有车能到的最大值。答案不会超过 C，增广次数不会超过 C。	时间复杂度:O(C(N+M)) 空间复杂度:O(N+M)
2013D	Factors	让我们用 f(k)表示 k 的质因子排列方案数，如 f(10)=2，f(20)=3。给你一个正整数 n，至少有一个 k 使得 f(k)=n，我们想知道最小的 k 是多少。n,k<2^63	x=∏ pi^ai的质因子排列方案就是$\frac{(\sum ai)!}{\prod ai!}$ 然后我们暴力搜索出所有满足条件的解，搜索的时候注意，随着 pi 的增加 ai 是单调不升的。	时间复杂度:O(N*logN) 空间复杂度:O(N) N 为搜索出来的 k 的数量。
2013E	Harvard	给定 b<=13 个内存，每个可以存放 m<=13 个变量，访问 0 号内存库代价是 1，你可以放置一个指针指向一个内存，放置或修改指针的代价为 1，你可以访问指针指向的内存代价也为 1，最开始这个指针不指向任何内存。给你一个操作序列，要求你给每个变量分配一个内存，最小化访问内存的代价。变量数 n<=13。	首先 0 号内存库必须要放满，我们就枚举将哪些变量放入 0 号内存库，然后预处理出来访问变量 i 的次数，访问变量 i 后访问变量 j 的次数 cnt。然后暴力搜索将剩下的变量放入哪个内存，放入的时候加上已经放入其他非 0 内存库的变量的次数 cnt。加上最优性剪枝。	时间复杂度:O(Cnm*search(ans)) 空间复杂度:O(length(seq))
2013F	Low Power	有 n 个机器，每个机器有 2 个芯片，每个芯片可以放 k 个电池。每个芯片能量是 k 个电池的能量的最小值。两个芯片的能量之差越小，这个机器就	二分最小的最大能量差，注意到任意两个芯片的最大值一定是相邻的，我们从大到小找到第一组满足条件的相邻的电	时间复杂度:O(NlogN) 空间复杂

		工作的越好。现在有 2nk 个电池，已知它们的能量，我们要把它们放在 n 个机器上的芯片上，使得所有机器的能量之差的最大值最小。	池，然后看看剩余的电池够不够放满剩下的机器。	度:O(N) N=2*n*k
2013H	Матрёшка	N<=500 个套娃排成一行，只有小的套娃能放到大的套娃内部，每次可以合并两个相邻的套娃，合并的代价是打开套娃的个数。求将 N 个套娃合并成若干个完整套娃的最小代价。完整的套娃是指套娃内部是从大小为 1 的套娃到到大小为 size 的套娃。	设 g[i][j]表示将 i 到 j 合并成一个套娃所需的最小代价。设 f[i]表示将前 i 个合并成若干个完整套娃所需的最小代价。 f[i]=f[j-1]+g[j][i]。问题在于如何求出 g 数组。考虑合并两个套娃的代价，a 是左侧套娃的最小值，b 是右侧套娃的最小值。如果 a<b 那么不需要打开的套娃数量就是左侧小于 b 的套娃的个数。我们设 k 是 i 到 j 中间最小的那个套娃所在的位置，先将 k 归到左侧套娃中，则左侧套娃内小于右侧套娃中最小值的数量单调不减，然后将 k 放到右侧套娃中，但是在用单调性维护的时候需要将 i 到 j 的套娃按升序排列，但是注意到有用的套娃大小<=N，所以我们采用计数排序做到 O(N)排序，这样我们可以 O(N^3)的求出 g 数组。而 F 数组的复杂度是 O(N^2)。	时间复杂度:O(N^3) 空间复杂度:O(N^2)
2013I	Pirate Chest	给一个深度不同的 N*M 的矩形水池，现在想向水池内放入一个长方体，要求底面宽<=a 底面长<=b，要求放到水池内后水面能淹没这个长方体，求最大化这个长方体体积。	设放入的长方体长宽高为 l、w、h。设长方体底部的深度为 d。 nmd+hwl=hnm 固定 d 发现 h 和 wl 成正比关系。所以我们希望 wl 尽可能的大。我们考虑枚举 w，枚举放置在哪个位置，d 取这些值中的最小值，然后用一个单调栈维护出来向左和向右能扩展到的最大的 l 值。	时间复杂度:O(N^3) 空间复杂度:O(N^2)
2013J	Pollution Solution	求一个点都在 x 轴之上的多边形，和一个以原点为圆心的上半个圆的交。	对多边形每条边与原点围成的三角形和半圆的交的有向面积求和。三角形和半圆的交分成三种情况考虑，没有交点，一个交点，两个交点。没有交点是求一个三角形面积，一个交点是求一个三角形加一个扇形面积，两个交点是求一个三角形和两个扇形面积。	时间复杂度:O(N) 空间复杂度:O(N)