

试题泛做

毛啸

题1

试题编号	Codeforces 273E
试题名称	Dima and Game
题目大意	算法讨论
<p>有n对整数(l_i, r_i) ($1 \leq l_i < r_i \leq p$)。每次可以选择满足$r_i - l_i > 2$的一对数(l_i, r_i)，将其替换为$(l_i + \lfloor \frac{r_i - l_i}{3} \rfloor, l_i + 2 \lfloor \frac{r_i - l_i}{3} \rfloor)$或者$(l_i, r_i - \lfloor \frac{r_i - l_i}{3} \rfloor)$。两人轮流操作，不能操作者输。</p> <p>给定n, p ($n \leq 1000, p \leq 10^9$)，求满足先手必胜的初始局面数量。</p>	<p>关键在于处理SG函数，$SG(l_i, r_i) = g(r_i - l_i)$，$g$定义为$g(1) = g(2) = 0, g(x) = \text{mex} \{g(\lfloor \frac{x}{3} \rfloor), g(x - \lfloor \frac{x}{3} \rfloor)\}$ ($x \geq 3$)。然后发现g分为值相同的很多段，然后发现对于一段$[l, r]$，满足$\lfloor \frac{x}{3} \rfloor \in [l, r]$的$x$范围是$[3l, 3r + 2]$，而满足$x - \lfloor \frac{x}{3} \rfloor \in [l, r]$的$x$范围是$[\lfloor \frac{3l-3}{2} \rfloor + 1, \lfloor \frac{3r}{2} \rfloor]$，我们将这两个区间加入两个队列$A, B$中，然后用二路归并的方法每次取出两个队列的队首取交集确定其g值加入答案队列中并pop其中一个元素。然后每次当两个队列中一个为空时，将答案队列中新加入的元素合并相同值后扩展出新的元素，处理完后可得知$SG = \{0, 1, 2\}$分别有几种情况。再用一个DP 求出n个SG 值异或和不为0的方案数即可。注意dp 的时候异或值有$\{0, 1, 2, 3\}$，而不仅是$\{0, 1, 2\}$!</p>
时空复杂度	空间 $O(d)$ ，时间 $O(n + d)$ ，其中 d 表示段数，在 $p = 10^9$ 时 d 大约为100。我猜想 d 为 $O(\log n)$ 级别，但无法证明。

题2

试题编号	Codeforces 317C
试题名称	Balance
题目大意	算法讨论
<p>给定一个由n个注水容器组成的系统。某几对容器由输水管道相连接。通过管道，你可以将整数升的水在其连接的容器间进行传输（管道都是双向的）。两容器间可能连着多于一根管道。管道的数量为e。每个容器的容积为v升。显然，在传输过程中任一容器中的水量不能超过v升。给定每个容器初始状态的水量a_i和目标状态的水量b_i，请求出一个输水方案实现这一目标。输水的总步骤数不能超过$2n^2$。</p>	<p>求出它的任意一棵生成森林T，使得每个联通块对应一棵树。如果当前存在一个度数为0的点，且它还不平衡，那么无解。否则找一个度数为1的点，如果它平衡了，那么就把它删掉。否则将它作为它所在的树的根，然后如果它容量多了，则尽量将它的孩子的容量向下赶，如果还没有空间往下传则无解，否则往下传并删除它；如果少了则全往上赶如果还不够传也无解；如果当前没有度数为0 或1的点则结束（实际上就是没有点了）。容易证明上述做法的正确性。</p>
时空复杂度	空间 $O(n + e)$ ，时间 $O(n^2 + e)$ 。

题3

试题编号	GCJ 2010 Final C	
试题名称	Candy Store	
题目大意	算法讨论	
有 k 位顾客依次光顾你的商店，每个人要买的糖果价值在区间 $[1, C]$ ($C \leq 10^{12}$)中且为整数。求最少事先所需要准备的糖果盒数量，使得其能保证无论顾客的需求如何，都可以恰好满足每位顾客。	一方面，若准备的 m 个盒子价值为 $a_1 \leq a_2 \leq \dots \leq a_m$ 。我们先得出答案的下界，如果所有 k 名顾客均购买 $a_i - 1$ 价值的糖果，而盒子仅有 a_1, a_2, \dots, a_{i-1} ，即 $\sum_{j=1}^{i-1} a_j \geq k(a_i - 1)$ ，即 $a_i \leq 1 + \left\lfloor \frac{1}{k} \sum_{j=1}^{i-1} a_j \right\rfloor$ 。另一方面，若每位顾客均买 C 价值糖果，则有 $\sum_{j=1}^m a_j \geq kC$ 。因此我们循环一遍求出最小的 m 使得上式成立即可。 容易证明上述算法的正确性，时空复杂度将在下面给出。	
时空复杂度	空间 $O(1)$ ，时间 $O(\text{Answer}) = O(k \log C)$ 。	

题4

试题编号	Codeforces 238D	
试题名称	Tape Programming	
题目大意	算法讨论	
有一个由数字和"i","j"构成的非空串，有一个指针，最开始指针的指向最左字符，移动方向为向右。如果指针指的位置是一个数字，输出这个数字，然后将指针沿着原来移动方向移动，同时将原来的数字减一。如果原来的数字为0则删除这个数字。如果指针指的位置是"i"或"j"，那么指针的移动方向对应的改为向左或向右，接着指针沿着新的移动方向移动。如果新的位置也是"i"或"j"，则删除原来的"i"或"j"字符。任何时刻如果指针指向了串外就结束。求每个子串执行这个过程之后输出每个数字的数量。	注意到任意时刻访问过的位置都是一个前缀，所以我们只需模拟一遍之后用一些数据结构维护即可。也就是访问到一个位置时所有以它开始的区间开始计数，以这个位置为结束位置的区间若未停止计数则停止计数(并统计答案)，若到了某些未停止计数的区间的左端点前面(这可以用堆维护)则那些区间全部停止计数。我们不能每次给每个区间计数，要利用前缀和的思想。	
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ ，注意往往有10的常数。本题存在复杂度更优的算法但这个复杂度足以通过本题。	

题5

试题编号	Codeforces 235E	
试题名称	Number Challenge	
题目大意	算法讨论	
求 $\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^c d(i \cdot j \cdot k)$ 。	可以通过差分法证明问题的答案即为 $\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^c \left\lfloor \frac{a}{i} \right\rfloor \left\lfloor \frac{b}{j} \right\rfloor \left\lfloor \frac{c}{k} \right\rfloor e(\gcd(i, j)) e(\gcd(j, k)) e(\gcd(i, k))$ ，其中 $e(x)$ 仅当 $x = 1$ 时为1，否则为0 因而我们枚举 i, j ，使得 $e(\gcd(i, j)) = 1$ ，那么 k 需要满足 $e(\gcd(i, k)) = e(\gcd(j, k)) = 1$ ，用莫比乌斯反演暴力即可过。我每次 $O(d(i)d(j))$ 便通过了本题。	

时空复杂度	设 $n = \max\{a, b, c\}$ ，空间方面本人预处理了一些东西因此是 $O(n^2)$ ，时间方面我暂时无法给出一个很好的上界，但实践证明改算法非常快。
-------	--

题6

试题编号	Codeforces 241D
试题名称	Numbers
题目大意	算法讨论
<p>您有$1, 2, \dots, n$的排列$a_1, a_2, \dots, a_n (1 \leq a_i \leq n)$。您想删除一些整数,使得结果序列满足以下三个条件:</p> <ol style="list-style-type: none"> 1. 由此产生的序列不是空的; 2. 序列中所有数异或和等于0; 3. 如果您把所有数按十进制从前往后依次无间隔地写在一行形成一个大的十进制数,这个数将会被p整除。 <p>给您序列和质数p,找到一种方法来满足上述条件。给定一个$1, 2, \dots, n$的全排列,从中划去任意项,使得剩下的数列(非空)满足:</p>	<p>我们可以设计一个动态规划来解决这个问题。</p> <p>令$f_{i,j,k}$表示前i个数,异或和为j,大的十进制数为k是否可行,很容易写出转移方程,这里不再赘述。</p> <p>事实上,对于很大的n,比如$n = 31$时,总是有解的,如果这个结论成立,由于我们只要求一组解,那么对于大于等于32的n,只需要保留它所有小于32的数形成的序列,从而将问题转化为$n = 31$的情形,即可解决问题。</p>
时空复杂度	时间复杂度 $O((\min\{n, 32\})^2 \times p)$, 空间复杂度 $O((\min\{n, 32\})^2 \times p)$

题7

试题编号	Codeforces 319E
试题名称	Ping-Pong
题目大意	算法讨论
<p>在这个问题中,每个时刻您都有一个区间的集合。您每次可以从集合中的区间(a, b)移动到另一个满足$c < a < d$或者$c < b < d$的区间(c, d)。您需要判断是否有一种从区间x到区间y的移动方案。区间(a, b)向(c, d)连一条有向边当且仅当$c < a < d$或$c < b < d$。支持两种操作(操作数量$\leq 10^5$):</p>	<p>我们用线段树维护区间,每个结点用数组维护(严格)包含该结点的区间,每次询问所有线段树中包含x或y的结点中的所有区间,并全部与该区间合并,之后将他们全部清空,加入很简单,直接将合并后的区间加入对应区间的数组中去即可,这样每个结点清空次数最多与加入次数相同,时间复杂度保证在了$O(n \log n)$。</p> <p>对于两个区间(a, b)和(c, d),如果它们被合并到了一个区间(在同一个区间集合)显然可以互相到达。</p> <p>否则如果还要互相到达,当且仅当(a, b)被(c, d)最后所在区间包含且不等。</p>
时空复杂度	时间 $O(n \log n)$, 空间 $O(n \log n)$ 。

题8

试题编号	Codeforces 286E
试题名称	Ladies' Shop

题目大意	算法讨论
<p>k种物品（每种物品有无限个）的质量分别为$1 \leq p_1 < p_2 < \dots < p_k \leq m$。任意取一个或几个物品组合可凑成的所有总质量中，不超过m者组成集合$A = \{a_1, a_2, \dots, a_n\}, a_i \leq m$。</p> <p>已知$m$和$a_1, a_2, \dots, a_n$，求出一组符合条件的$p_1, \dots, p_k$并使得$k$最小，或者判断无解。</p>	<p>容易证明，无解当且仅当$\exists i, j, a_i + a_j \leq m, a_i + a_j \notin A$。若有解，则对于某个质量$a_x$，若$\forall i, j, a_x \neq a_i + a_j$，则必然存在某个$p_y$等于$a_x$，反之每个$p_y$都存在一个$a_x$满足上述条件。易发现需要统计的内容具有卷积的形式，用FFT即可。如果您不信任FFT的精度，由于卷积后每个数不超过n，您可以在足够大的质数意义下使用NTT，如使用$479 \times 2^{21} + 1 = 1004535809$，这个数的一个原根为3。</p>
时空复杂度	时间 $O(m \log m)$ ，空间 $O(m)$ 。

题9

试题编号	Codeforces 306C
试题名称	White, Black and White Again
题目大意	算法讨论
<p>在n天内发生了两两不同的w件好事和b件坏事，而且每天至少发生一件事，且同一天发生的事是要计顺序。n天按顺序被分成3个非空的部分，满足第一、三部分只发生好事，且第二部分只发生坏事。求总方法数模$10^9 + 9$的值。</p>	<p>设$way_{i,j}$为j天发生i件事的方案数，则$way_{i,0} = [i = 0]$，且有$way_{i,j} = j \times way_{i-1,j-1} + way_{i-1,j} \times (i-1+j) (j > 0)$。答案为$\sum_{i=2}^{n-1} way_{w,i} \times (i-1) \times way_{b,n-i}$。</p>
时空复杂度	空间 $O(n^2)$ ，时间 $O(n^2)$

题10

试题编号	Codeforces 274C
试题名称	The Last Hole!
题目大意	算法讨论
<p>有n个圆，开始半径为0，然后每个时刻半径增长一个单位，中间会形成“洞”，判断是否有洞生成，若有洞生成求出最后一个洞消失的时刻。</p>	<p>洞只有可能是锐角三角形的内心或者正方形的对角线交点，求出来以后判断与其他点的距离是否均不小于到对应多边形的端点的距离即可。</p>
时空复杂度	时间 $O(n^4)$ 且常数十分小，空间 $O(n)$ 。

题11

试题编号	Codeforces 301C
试题名称	Yaroslav and Algorithm

题目大意	算法讨论
<p>考虑这样一个算法：</p> <ol style="list-style-type: none"> 1. 这个算法接受一个字符串作为输入。我们设这个输入字符串为a。 2. 这个算法由一些命令组成。i号命令的形式为"$s[i]>>w[i]$"或"$s[i]<<w[i]$"，其中$s[i]$和$w[i]$是长度不超过7的字符串（可以为空），由数字或字符"?"组成。 3. 这个算法每次寻找一个编号最小的命令i，使得$s[i]$是a的子串。如果没有找到这样的命令，那么整个算法终止。 4. 设找到的命令编号为k。在字符串a中，$s[k]$第一次出现的位置会被$w[k]$替换。如果这个命令形如"$s[k]>>w[k]$"，那么这个算法继续执行（译注：回到第3步）。否则，算法终止。 5. 算法的输出就是算法终止时字符串a的值。 <p>给定一个n个正整数的集合，求一个算法使每一个数加1。</p>	<p>可以证明这样的算法即可解决问题：</p> <p>0??<>1 1??<>1 ... 8??<>9 9??>>??0 ??0<>10 ?0>>0? ?1>>1? ... ?9>>9? ?>>?? >>? 直接输出即可。</p>
时空复杂度	时间空间复杂度显然均为 $O(1)$ 。

题12

试题编号	USACO Dec 12
试题名称	Gangs of Istanbul/Cowstantinople
题目大意	算法讨论
<p>给定一个序列和一个栈，要求您选定一个顺序将序列中的数加入栈中，如果与栈顶不同则两个数均消失，否则保留(栈空的话直接加入)，求一个字典序最小的方案最大化最后剩下1的个数。</p>	<p>可以利用鸽巢原理和剩下数和奇偶性直接算出最后剩下1的个数，那么最后牺牲的1统一加到最前面，接着每次如果能加入则加入最小值，否则加入当前最多的数(中最小的)，能加入就是加入那个数之后剩下的最多的数如果马上全部加入，那么后面加入的数可以将这个数全部抵消，所以关键是维护最大值，注意最大值只有$O(n)$级别，并且每个数的个数只会减少，减少次数也只有$O(n)$次，所以可以$O(n)$维护，注意如果不能加入那么最大值个数不会超过2，所以只要实现得好，时间复杂度可以保证，本题细节较繁琐，这里不一一赘述。</p>
时空复杂度	时间 $O(n)$ ，空间 $O(n)$ (用手写堆 $O(n \log n)$ 常数小点也可以通过本题)。

题13

试题编号	Codeforces 293E
试题名称	Close Vertices

给定一棵树，求满足距离不大于 L ，且最短路径上边权和不超过 W 的点对数。	树分治，对于固定根，可以用补集转化的思想，转化为任意两点的个数减去所有同一子树中点的个数，具体计算方法可以从大到小枚举深度，可行的深度将越来越大，维护可行的深度中的点到根的权值和形成的集合，每次相当于加一个元素到集合中，或者询问集合中比某个元素小的值有多少个，显然可以用平衡树，但简单起见可以离散化+fenwick树解决。
时空复杂度	时间 $O(n \log^2 n)$ ，空间 $O(n)$ 。

题14

试题编号	Codeforces 331E2
试题名称	Deja Vu
题目大意	算法讨论
给定一张有向图，每条边上有许多编号，一条路径合法，当且仅当，这条路径上的点构成的序列，和这条路径上的边上的编号构成的序列完全一样。要求统计长度为 $1, 2, \dots, 2n$ 的合法路径个数。	我们先求出对每条出现在自己编号当中的边(称之为“基础边”)，它扩展出的最短的满足条件的路径，称为这条边的“基础路径”，可以证明每条基础路径中必然只含有一条基础边，接着我们进行一个DP，对于一条基础路径，它可能延伸一个头和尾，其中不含其它基础边，也可以将两条仅含基础边的路径用一个不含编号的边合并，用 $dp[i][j][k]$ 表示长度为 i 的从 j 到 k 的合法路径的个数，很容易设计出DP方程，为了减少时间复杂度，为了在较低时间复杂度内进行最后一种情况的转移，可以在前面的情况处理好之后用 $g[i][j]$ 表示长度为 i 的以 j 结尾的合法路径个数再进行转移。
时空复杂度	时间 $O(n^4)$ ，空间 $O(n^3)$ 。

题15

试题编号	Codeforces 305E
试题名称	Playing with String
给定一个字符串集合，每次可以选择一个字符串，我们称上面写着的字符串为 t 。注意一开始时集合中只有一个给定的串。接着选择一个 $i(1 \leq i \leq t)$ 使得存在一个正整数 $k(0 < i - k, i + k \leq t)$ 满足 $t_{i-1} = t_{i+1}, t_{i-2} = t_{i+2} \dots t_{i-k} = t_{i+k}$ 然后这个字符串分成了3份： $t_1 t_2 t_3 \dots t_{i-1}, t_i, t_{i+1} t_{i+2} t_{i+3} \dots t_{ t }$ 。两个人轮流行动，不能操作的人输。问是否先手必胜，若必胜求第一步能选择的最小 i 。	我们可以枚举 i ，所以问题关键在于算一个局面的 sg 函数值，将所有满足 $t_{i-1} = t_{i+1}$ 的 i 提取出来，那么分为了很多段，每一段是独立的子游戏，所以问题关键在于求一段的 sg 值，后者很容易通过递推推出来。
时空复杂度	时间 $O(n^2)$ ，空间 $O(n)$ 。

题16

试题编号	CODEFORCES 314E
试题名称	Sereja and Squares

题目大意	算法讨论
给定 n 个点，第 i 个点在 $(i, 0)$ ，有些位置是 $?$ ，有些位置已填小写字母，求有多少种方法给每个位置填一个不为 x 或 X 的大写或小写字母，使得它可以完全分成不相交的点对，使得每对点中横坐标小的是小写字母，横坐标大的是对应的大写字母(如 a 和 A)，以每对点为对角线作正方形，则正方形不相交或接触。	很容易转化成类似括号序列的问题。设 $dp_{i,j}$ 表示前 i 位小写比大写多 j 个的方案数，则若第 i 位为 $?$ 则 $dp_{i,j} = dp_{i-1,j-1} * 25 + dp_{i-1,j+1}$ ，否则 $dp_{i,j} = dp_{i-1,j-1}$ 。注意到可以滚动数组，空间无压力，但是时间复杂度大。但这题可以卡常数！注意到 i, j 的奇偶性不同(或相同，取决于实现细节)时 $dp_{i,j} = 0$ ，所以可以卡去2的常数，而 $j > i$ 时显然为0，又卡去2的常数，然后在 $O2$ 和 $4s$ 的情况下，神奇的过了(如果恰好 $4s$ 出解，每秒运算次数达到六七亿次，而计算几何等题目几百万次都有时候达不到，所以常数对运行时间的影响是很大的)。
时空复杂度	时间 $O(n^2)$ ，常数很小，空间 $O(n)$ 。

题17

试题编号	Codeforces 338D
试题名称	GCD Table
题目大意	算法讨论
给定一个 n 行 m 列的矩阵 $a_{i,j}$ ，其中 $a_{i,j} = gcd(i, j)$ ，求序列 s 是否在矩阵的一行中作为连续子序列出现。	容易看出若有解则 $n = lcm(s)$ 时也有解，接着我们发现每一个 s_i 意味着一个同余方程式，容易列出一个同余方程组，但应用 CRT 时会遇到问题，但我们可以对每个质数的幂列任意一个方程，解出一个解，若这个解不满足条件或者超出范围则无解，否则有解，注意解出来在模意义下为0则必须取 $lcm(s)$ 。
时空复杂度	时间 $O(n^{0.5} + m \log n)$ ，空间 $O(m)$ 。

题18

试题编号	USACO March Contest 2008, Gold Division, Problem 1
试题名称	Land Acquisition
题目大意	算法讨论
给定 n 对数 a_i, b_i ，求将他们分成若干个子集，每个子集的权值为 $\max\{a\} \times \max\{b\}$ ，求一种划分方案最小化所有子集的权值和。	将 a_i 升序排序，那么答案可以证明一定是取一段一段的子序列，所以容易设计一个 $O(n^2)$ 的DP，然而我们要优化，但我们注意到每次最优决策是在 b 的后缀最大值处，并且DP的转移可以斜率优化，两者结合之后显然可以用可持久化线段树记录每个时刻的凸壳，或者利用分治。分治的做法是左边的按左右顺序依次加入所有后缀最大值，并且每次用当前最优值求出右边的值大于左边当前值的位置的DP值。本题有简单得多的 $O(n \log n)$ 做法，以上纯属笑谈。
时空复杂度	时间 $O(n \log^2 n)$ ，空间 $O(n)$ 。

题19

试题编号	Codeforces 323B(Testing Round #7)
试题名称	Tournament-graph

构造一个有 n 个结点的竞赛图，使得对任意两个结点 u 和 $v(u \neq v)$ ，从 u 到 v 的最短距离不超过2。	在题目范围内只有 $n = 4$ 时无解，其他情况均有解。构造出 $n = 3$ 和 $n = 6$ 的答案，然后每次选两个点 a, b ，连一条 $a \rightarrow b$ ，让所有前面的点 x ，连一条 $x \rightarrow a$ 和 $b \rightarrow x$ 即可。
时空复杂度	时间 $O(n^2)$ ，空间 $O(1)$

题20

试题编号	Codeforces 342D
试题名称	Xenia and Dominoes
题目大意	算法讨论
一个 $3 \times n$ 的棋盘，有一些障碍，和一个指定留空的格子。把剩下的格子全部用 1×2 骨牌铺满，使得存在一个骨牌可以移动。问方案数模 $10^9 + 7$ 。 $n \leq 10000$	空格周围至多只有4个方向能够放下骨牌并且指向它，可以用 2^4 容斥原理解决，然后转化为经典的状态压DP。
时空复杂度	空间 $O(n \times 2^3)$ ，时间 $O(2^3 \times 4^3 n)$ 。

题21

试题编号	Codeforces 319D
试题名称	Have You Ever Heard About the Word?
题目大意	算法讨论
给定一个字符串，每次去掉最短最前的两个连续的相同的串中的一个，求最终串。	枚举长度 l ，然后根据第 i 位与第 $i + l$ 位的最长公共前后缀可以算出这里是否有重复，可以用 $hash$ 算出这个值，每个长度做完后，如果字符串有改变则重构 $hash$ 数组，注意每次重复串去掉长度减少 l ，所以最多重构 $\sqrt{ S }$ 次。本题还有可以用平衡树维护 $hash$ ，后者虽然难写但它严格正确且时间复杂度渐进意义下更优。
时空复杂度(用 $hash$)	设 $n = S $ ，时间 $O(n\sqrt{n} + n \log^2 n)$ ，空间 $O(n)$ 。

题22

试题编号	Codeforces 241B (Bayan 2012-2013 Elimination Round)
试题名称	Friends
求 n 个数的两两异或值的前 m 大和。	设数的最大值为 A 。首先在 $trie$ 上 $O(n \log A)$ 二分出第 m 大(类似线段树求区间 k 大)，在CF原题上用 $O(n \log^2 A)$ 二分即可。然后通过利用 $trie$ 记录每个子树每一位为1的个数，即可统计出答案，这一步时间复杂度为 $O(n \log A)$ 。本题的一个扩展是假设 A 很大，那么我们用 $trie$ 不太好，这时我们可以考虑用压缩型字母树，既可以省空间也可以省时间(时间复杂度上一个因子由 $O(\log A)$ 变为 $O(\log n)$)，但压缩型字母树十分难写，注意到本题是离线建树，所以可以排序后通过寻找当前区间当前位第一个1的位置的方式方便地建出压缩型字母数，敏锐的同学可能发现其实这种做法完全可以脱离 $trie$ 而存在，因为排序后子树查询变成区间查询，因此如果不管常数，子树信息可以少记录很多而采用前缀和的方式查询，但渐进复杂度得不到显著优化。值得一提的是如果 A 很小(如在 10^5 范围内)，但 n 很大，显然可以用 fwt 实现一个时间复杂度为 $O(A \log A)$ ，空间复杂度为 $O(A)$ 的算法。

时空复杂度	时间 $O(n \log^2 A)$ 或 $O(n \log n \log A)$, 空间 $O(n \log^2 A)$ 或 $O(n \log A)$ (压缩型字母数节点数 $O(n)$)。
-------	---

题23

试题编号	Codeforces 303D	
试题名称	Rotatable Number	
题目大意	算法讨论	
一个数被称为可旋转数, 当且仅当所有从它通过旋转得到的所有不同的数就是它自己乘以从1 到数的长度得到的数, 旋转一个数就是将它的最后一位数字放到最前面, 并且允许有前导0。求最大的 $b(1 < b < x)$, 满足在b进制下存在一个长度为 n 的正“可旋转数”(允许有前导零)。	<p>可以证明这种数乘以$n + 1$之后是某个b的次幂减1, 所以有</p> $b^n - 1 \equiv 0 \pmod{n + 1} \quad (1)$ $b^n \equiv 1 \pmod{n + 1} \quad (2)$ $b^x \not\equiv 1 \pmod{n + 1} \quad (0 \leq x < n) \quad (3)$ <p>容易发现$\varphi(n + 1)$要是n的约数, 因此$n + 1$是质数, 所以用找原根算法即可, 也就是从x开始从大到小枚举, 每次检查那个数的n 的约数次方是否为1即可。</p>	
时空复杂度	时间 $O(kd(n))$ 其中 k 为满足条件的最大 b 与 x 之差, $d(n)$ 为 n 的约数个数, 空间 $O(1)$ 。	

题24

试题编号	Codeforces 351D	
试题名称	Jeff and Removing Periods	
题目大意	算法讨论	
对一个长度为 n 的序列 a_1, a_2, \dots, a_n , 你可以选择三个整数 $v, t, k(1 \leq v, t \leq n; 0 \leq k; v + tk \leq n)$, 要求满足 $a_v = a_{v+t} = a_{v+2t} = \dots = a_{v+t(k-1)} = a_{v+tk}$ 并将 $a_v, a_{v+t}, \dots, a_{v+tk}$ 删除。并将剩下的数重新排列, 定义一个数列 a 的美好度为将所有数字全部删除需要的最小步数。多次询问序列的连续子序列的美好度。	<p>我们只要删除一次, 再把序列排序, 就可以在不同的数字个数步内完成, 所以答案为不同的数字个数或不同的数字个数减一。</p> <p>不同的数字个数可以用莫队维护每个数出现次数解决。减不减一取决于序列中存不存在一个出现过的数满足它出现的位置构成等差数列, 这个可以通过预处理每种数的每个出现位置开始, 要满足这种数的出现位置构成等差数列, 所能够到的最远位置, 预处理很简单但容易错, 如果您清橙上拿80分, 那么应该就是预处理出了点问题(至少在我交题的时候是这样)。然后每次莫队移动区间时, 维护每种数的首次出现位置, 移动完毕后对于移动中涉及到的所有数根据它的首次出现位置所能够到的最远位置与区间终点的比较来判断它是否是等差数列。</p>	
时空复杂度	时间 $O(n^{1.5})$, 空间 $O(n)$ 。	

题25

试题编号	Codeforces 277D	
试题名称	Google Code Jam	

题目大意	算法讨论
有 n 道题，每道题有一个简单部分和难的部分，必须先做简单部分才能做难的部分，简单部分不可能 fst ，难的部分可能 fst ，给定每道题做每个部分的时间，和做出并且不 fst 之后的得分，和每道题难的部分 fst 的概率，最大化期望得分和该期望得分下最小的罚时(罚时是最后一次AC的时间)。	给定做的部分最优顺序一定是先做所有简单部分再按 $\frac{fstprob_i \times scoredifficult_i}{1 - fstprob}$ 从小到大的顺序做难的部分，因此按这个排序后记录背包即可，注意 dp 要同时记录两个量。
时空复杂度	时间 $O(nt + n \log n)$ ，空间 $O(nt)$ 。

题26

试题编号	Google Code Jam 2009 Final A	
试题名称	Year of More Code Jam	
题目大意	算法讨论	
给定一些序列，每个序列在大序列中每个位置等概率的开始，且每个序列相邻元素的位置之差不变，每个位置可能有多个元素，若有 S 个元素则产生 S^2 的权值，求大序列的期望总权值。	对于每个位置，某个序列可能在这个位置有元素的概率是在该序列从大序列头开始时位置不在这个位置之后的元素个数除以大序列长度，设 x_i 表示序列 i 在这个位置是否有元素， p_i 表示有元素的概率，则这个位置对总权值的贡献为 $E[(\sum_{i=1}^n x_i)^2] = \sum_{i=1}^n E[x_i^2] + \sum_{i=1}^n \sum_{j=1}^n E[x_i \times x_j]$ 。右边第一项就是 $\sum_{i=1}^n p_i$ ，第二项就是 $\sum_{i=1}^n \sum_{j=1}^n p_i \times p_j$ 。而由于当当前位置不小于所有序列从大序列头开始时元素的最大位置时，这个期望不变，所以简单乘一下即可。易知分母不会爆 $long\ long$ ，因此毋需高精度。	
时空复杂度	设 n 为所有序列中的总元素个数， d 为最大元素位置，则复杂度为时间 $O(n + d)$ ，空间 $O(n + d)$ 。	

题27

试题编号	Codeforces 321D	
试题名称	Ciel and Flipboard	
题目大意	算法讨论	
给点一个 $(2x - 1) \times (2x - 1)$ 的棋盘，格子上的权值，每次可以将一个 $x \times x$ 的子矩形取相反数，最大化矩形内的数字和。	设 $a_{i,j}$ 表示第 i 行第 j 列最后是否被取反，那么有 $a_{i,j} \otimes a_{i,x} \otimes a_{i,x+j} = 0$ ， $a_{i,j} \otimes a_{x,j} \otimes a_{i+x,j} = 0$ 。枚举 $a_{i,x} (1 \leq i \leq x)$ 的值，则 $a_{i+x,x} = a_{x,x} \otimes a_{i,x} (1 \leq i < x)$ ，故可以确定 $a_{i,x} (1 \leq i \leq 2x - 1)$ ，对于每个 $j (1 \leq j < x)$ ，若 $a_{i,j}$ 确定则 $a_{i,j+x}$ 确定，因而若 $a_{x,j}$ 确定则 $a_{i+x,j}, a_{i+x,j+x}$ 确定，所以可以算出 $a_{x,j}$ 的最优值，从而得到此时的答案。	
时空复杂度	时间 $O(2^x \times x^2)$ ，空间 $O(x^2)$ 。	

题28

试题编号	USACO Dec10 Gold	
试题名称	Threatening Letter	

题目大意	算法讨论
给定两个字符串 S_0, S_1 , 求将 S_1 分成尽量少的连续子串, 使得每个子串也是 S_0 的子串。	对 S_0 建后缀自动机, 然后对 S_1 跑一遍, 若不能走则回到父亲, 即可求出每一位往左匹配最远能到什么地方, 显然贪心地直接让每一位匹配得最远即可。
时空复杂度	时空复杂度均为 $O(S_0 + S_1)$ 。

题29

试题编号	GCJ 2009 Final C
试题名称	Doubly-sorted Grid
题目大意	算法讨论
给定一个矩阵, 上面有些地方填了字母, 用字母填完剩下的地方并使得每个地方的字母不大于其右边和下边的位置填的字母(如果有), 求方案数。	挨个挨个字母填, 每个字母从上往下, 从左往右填, 则每个字母填完后(任意时刻也是), 轮廓线必须是一个只能往左或往下的折线, 这样的折线个数不多, 只有 $\binom{20}{10}$ 种, 预处理出所有折线在每一行填一个字母的转移之后, 很容易设计出 DP 方程, 这里不再赘述。
时空复杂度	时间 $O(\sigma \times \binom{2n}{n} \times n)$, 空间 $O(n \times 2^{2n})$ (本人的程序)。

题30

试题编号	USACO Open 07
试题名称	Connect
题目大意	算法讨论
一个 $2 \times n$ 的点阵, 每次可以删边加边, 边只能连接相邻的点, 问两点间是否存在一条路径, 使得路径只会只往左不往右或者只往右不往左。	直接线段树维护连通性即可, 区间 $[i, j]$ 表示第 i 列与第 j 列间(包括端点)的连通性。细节较繁琐。具体可以设一个量表示某一列的竖边是否存在, 合并两个子节点的信息时利用这个量合并即可。另外本题是BZOJ1018的弱化版加上强制在线, 而BZOJ1018加强制在线也可以做, 不加的话似乎可以利用动态图问题的通用方法cdq分治, 强制在线不考虑时间效率可以用WC上讲的动态维护图的连通性的方法做,
时空复杂度	时间 $O(n \log n)$, 空间 $O(n)$ 。

题31

试题编号	Codeforces 261E
试题名称	Maxim and Calculator

题目大意		算法讨论
<p>有两个整数单元，一开始第一个单元包含数字1，第二个单元包含数字0。进行两种操作：</p> <ol style="list-style-type: none"> 1. 假设第一个单元的数字为a，第二个单元的数字为b，那么将第二个单元的数字改成$b + 1$。 2. 假设第一个单元的数字为a，第二个单元的数字为b，那么将第一个单元的数字改成ab。 <p>给定l, r，有多少个$x \in N^+ \cup [l, r]$满足，存在一种方式从计算器初始状态开始，操作不超过p步之后使得第一个单元中的数字为x。</p>		<p>显然该数要没有大于p的素因子，在p, r取最大值100和10^9时，这样的数也不多(见时空复杂度)，因此可以求出所有这样的数，然后用一个dp求出得到任意数的最少操作次数，再统计答案。</p>
时空复杂度	时间 $O(d \times p)$ ，空间 $O(d)$ ，其中 d 为不大于 r 的没有大于 p 的素因子的数的个数，在题目范围内最多不超过 3×10^6 。	

题32

试题编号	GCJ 2009 Final E	
试题名称	Marbles	
题目大意		算法讨论
<p>给定一个序列，每个位置的元素匹配恰好匹配另一个位置的元素，求将序列划分成两个集合，使得每个集合内的元素可以根据匹配关系形成括号序列，且最小化括号序列的最大层次。</p>		<p>根据矛盾关系建图，一定构成二分图否则无解，形成一些连通块，可以发现连通块之间要么没有公共部分要么包含所以关系构成一棵树，每个连通块可以朝上或朝下，用(节点编号, y坐标的最大值, 该节点的朝向)作为状态，最小的y坐标(或其绝对值)作为值，进行树形DP即可，具体方法不再赘述。</p>
时空复杂度		

题33

试题编号	Codeforces264D	
试题名称	Colorful Stones	
题目大意		算法讨论
<p>两个字符串，两个指针指向串的开始，如果两个指针所指位置相同则必须同时向后移一位，否则可以将任意一个向后移一位，求可能的不同状态个数。</p>		<p>求出对于第一个串中每个位置，第二个串的最早位置和最后位置(若超出则记为最后一个位置)，每次答案加上中间的位置个数，对于当前位置的最后两位字符c_0c_1，若$c_0 \neq c_1$则要减去第二个串中所有的最后两位字符是c_1c_0的位置个数，用前缀和统计即可。</p>
时空复杂度	时空复杂度均为 $O(S + T)$ 。	

题34

试题编号	codeforces 339E	
试题名称	Three Swaps	

题目大意	算法讨论
给定一个 n 的全排列，它由 $1, 2, 3, \dots, n-1, n$ 经过三次区间翻转得到，求一种可能的翻转方式(也必须是3步以内)。	只有使得翻转后开头与其前或结尾与其后仅差一（或翻转后区间第一个变为1或最后一个变为 n ）的方式才有用，每次枚举就可以通过本题。
时空复杂度	时间方面我只知道是 $O(n^3)$ 但很快，不知是否有更好的界，空间是 $O(n)$ 。

题35

试题编号	codeforces 249D	
试题名称	Donkey and Stars	
题目大意	算法讨论	
给定一些点，和两个正的斜率，每个点可以走到它右上角的与他之间的斜率在两个斜率之间的点，求最长的走的步数。	可以用叉积求出每个点到两个斜率向量的投影，然后转化为直角坐标系上的问题，即为经典的LIS问题。	
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。	

题36

试题编号	codeforces 309D	
试题名称	Tennis Rackets	
题目大意	算法讨论	
给定一个正三角形，每条边上有 n 个小孔，把该边分成了等距的 $n+1$ 段。求选三个点，使得首先每条边上，离每个顶点最近的 m 个不能选，然后三个点需要连成一个钝角三角形，且三个角分居正三角形框架三条不同的边上。求有多少种可行的方案。	枚举两个点，显然另一条边上能选的点是一段区间，且若枚举的第二条边上的点是按从距离另一条边从近到远的顺序，则区间不断缩短，可以用两个指针维护区间，每次用叉积判断，这样复杂度高，需要卡常数才可以通过本题。	
时空复杂度	时间 $O((n-2m)^2)$ ，空间 $O(1)$ 。	

题37

试题编号	USACO Open 14	
试题名称	Code Breaking	

题目大意	算法讨论
给定一棵树，每个点可以给一个0-9的权值，要求从某些点到根的路径上的前5个点的权值的序列不为给定的序列，求不合法的方案数。	容易设计一个 dp ，以(点编号，该点往上的4个数的情况)为状态，以该状态下该点的子树的合法方案数，进行 dp ，那么我们枚举这个点的值，然后将子树的对应 dp 值乘起来即可。但这样状态过多会超时。我们注意到有用的状态(即可能对方案数造成影响的状态)只有 $5n$ 个，所以可以优化，但暴力转移会是 $O(nm)$ ，因为一个点可能有 m 个状态，度数又为 n ，但有用的(往上的3个数,该点的数)(即可能对某一个子树的转移造成影响)的组合是 $O(n)$ 的，我们可以将这样的4个数组合即其所有前缀加入一个集合中，然后对集合进行dfs，每次可能一个子树的转移会进行变化，由于是乘可以用线段树维护(直接维护成绩似乎可行，但是原数不是质数，存在一些比较麻烦的问题)，然后对每个状态枚举该点填的数，该状态的答案加上最长的在集合中的前缀对应的转移值即可。可以证明这样复杂度是 $O(n \log n)$ 的。
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。

题38

试题编号	Codeforces 280E	
试题名称	Sequence Transformation	
题目大意	算法讨论	
给定一个序列 x_1, x_2, \dots, x_n ，给定正整数 q, a, b ，求另一个序列 y_1, y_2, \dots, y_n ，使得 $a \leq y_{i+1} - y_i \leq b, 1 \leq i \leq n-1$ ，且 $1 \leq y_1, y_n \leq q$ ，保证对所有 $\forall i, 1 \leq x_i \leq q$ ，且 x 序列单调不递减。最小化 $\sum_{i=1}^n (x_i - y_i)^2$ 。	用 $dp_{i,j}$ 表示前 i 个数， y_i 数取 j 的答案，这样状态数是无穷的，注意到状态是关于 y_i 的函数，且其导数单调不递减，所以我们可以维护其导数，每次求出导数为零的点，并根据它进行转移，具体就是它左边向右移动 a 个单位，右边向右移动 b 个单位，中间形成一个长度为 $b-a$ 的平台。我们用平衡树维护导数，利用平衡树的基本操作可以实现以上过程。求出每一个位置的最值之后，很容易求出最优解。本题细节较多，不一一赘述。	
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。	

题39

试题编号	CODEFORCES 317E	
试题名称	Princess and Her Shadow	
题目大意	算法讨论	
有一个方块组成的地图，有一些方块上是障碍，有两个方块 A 和 B ，每次可以往某个方向移动使得 A 可以移动，并且若 B 可以移动则 B 也跟着移动，判断 A 能否和 B 重合，若能则给出方案。	若两个不连通则不能。否则一定能，如果二者均不与外界联通，则求出 A 到 B 的最短路，用一个队列维护，若 B 也移动了则将该移动方向加入队尾，每次移动队头的方向，若二者重合则找到答案，否则我们假设 A 在 B 的左下方，则先将 A 移动到左下角，并向下移动很多步，再将 B 移动到左下角(这时候 A 永远在空地上，直接走最短路就行了)，然后找到最下的(如果有多个则找其中最左的)方格，先将 B 的横坐标与之对齐，再不断向上移动直到 B 被卡住并且 A 与 B 纵坐标相同，然后向左移动一次，向上再移动一次，再不断向右直到重合为止。	

时空复杂度	时间 $O(step + size)$, 空间 $O(step + size)$, $step$ 指产生的答案步数, $size$ 指棋盘大小(实际上方便起见至少要开很多倍的棋盘大小的内存, 时间也一样)。
-------	---

题40


试题编号	usaco2009 open gold
试题名称	Tower of Hay
题目大意	算法讨论
给定长为 n 的数组 $a[]$, 将其分为 h 段, 前一段之和不小于后一段, 求 h 的最大值。	可以证明, 使得第一层最小的方案中必定存在能使段数最大的方案, 用 $f[i]$ 表示 $a[i \cdots n]$ 这一段中, 底层的最小长度。 $f[i] = \min_{i < j \leq n+1, \text{sum}[i, j-1] \geq f[j]} \{\text{sum}[i, j-1]\}$ ”。这个 DP 可以用线段树或者单调队列优化, 这里不再赘述。
时空复杂度	时空均为 $O(n \log n)$ (使用线段树)或者 $O(n)$ (使用单调队列)。

题41

试题编号	Google Code Jam 2011 World Finals C
试题名称	Program within a Program
题目大意	算法讨论
数轴上整数点处有一个电线杆, 初始颜色为0。您有一个初始状态, 您只能知道您的状态和这个位置的电线杆的颜色, 并根据这个决定往哪边移动或者停止。每种状态和颜色的二元组只能对应有一种决定, 且过程中出现过的二元组的必须有一种对应决定。求构造30个以内的二元组与决定的组合, 使得该组合满足上述两个条件且您从原点开始按照这个组合转移最终能够到 n 号点终止。	大致思路就是先讲 n 的二进制填上去(0变为1, 1变为2), 然后每次先从右往左扫, 将二进制数减一, 并在此之后从左往右扫, 将所有二进制位右移一位, 直到二进制数变为零为止, 首先填二进制可以用位数个单独的状态做, 然后从右往左扫可以用两个状态, 遇到过2, 那么就不用管了, 没遇到过2, 那么1变为2, 2变为1并转移到遇到过2, 从左往右可以用两个状态, 前一位是1, 前一位是2, 再考虑一些细节, 经过一些小的思考不难在规定条件内实现整个过程。
时空复杂度	时空均为 $O(1)$ 。

题42

试题编号	Codeforces 329D
试题名称	The Evil Temple and the Moving Rocks

题目大意	算法讨论
一个 $2k \times 2k$ 的矩阵里有很多块石头，每种石头可以往一个方向移动。您首先可以任意放石块(不必放满)。之后您可以选择其中一块石头激活。被激活的石头将会一直朝着它的方向移动，直到撞到了其他石块或者撞到了房间四周的围墙（如果在它的方向上紧挨着就有其他石块，它将不会有任何移动）。之后这块石头将停止运动。如果它撞到了围墙，则游戏结束。否则，它所撞击到的石块将被激活，且这一过程将会持续发生。倘若石块在撞击到围墙或者其他石块之前至少移动了一步，则该次撞击将会发出响声。给定 k, x ，求一种布置方案使得响声个数大于等于 x 。	类似这样构造即可通过本题，例如当 $k = 9$ 时： 
时空复杂度	我的程序是时间 $O(k^2)$ ，空间 $O(k)$ 。

题43

试题编号	Codeforces 269D
试题名称	Maximum Waterfall
题目大意	算法讨论
平面上有一些板子，水可以从第 i 块板流向第 j 块板当且仅当二者的水平位置存在相交部分，且第 j 块板在第 i 块板下方，并且不存在高度在 i, j 之间的板子 k 使得 i 与 k 和 k 与 j 均满足以上两个条件。此时 i 到 j 的流量为二者的水平相交部分的长度。水会沿着一条单向的路径从顶部流到底部，称为“瀑布”。如果水流到了一块水平板上（除了墙的底部），水会流向恰好一个更低的水平板。整个瀑布的水流量被定义为水流路径上每连续的两块水平板间的水平相交部分长度的最小值。求最大宽度的瀑布。	用扫描线，按每块板子左边的开始位置的递增顺序依次加入板子，每次将它的上边的和下边的两块板子之间的连边拆开，并且均与它连新边，最后从下到上做一次DP即可。
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。

题44

试题编号	GCJ 2013 Final E
试题名称	Let Me Tell You a Story
题目大意	算法讨论
给定一个序列，求有多少种删数的方案(顺序不同算一种)，使得在删除操作未结束时留下来的数列不是单调不下降的，且最后留下来的数列是单调不下降的。有 T 次询问。	求出共有多少种长度为 $l, 1 \leq l \leq n$ 的序列，记答案为 cnt_l ，然后对 $cnt_l \times (n - l)! - (l + 1) \times cnt_{l+1} \times (n - l - 1)!$ 求和即可，前者可以用树状数组在 $O(n^2 \log n)$ 的时间复杂度下算出。
时空复杂度	时间 $O(T \times n^2 \log n)$ ，空间 $O(T \times n^2)$ 。

题45

试题编号	codeforces 264E
试题名称	Roadside Trees
题目大意	算法讨论
给定一个长度为 n 的序列，每次可以在一个位置插入一个数，也可以删除从左往右一个数，满足插入的数和删除的数从左往右的编号均不大于10，求维护每次所有当前数的最长上升子序列，保证任意时刻数均不相同。	用很多的 set ，维护按位置，按数大小排列的当前数，和某个 DP 值的按位置排序的所有值。然后首先显然对于某个 DP 值的所有位置从左往右不递减，然后对于插入，每次询问它右边第一个数的高度并与自己比较即可，同时很容易证明这个问题可以对 DP 值二分。对于更新 DP 值无论如何都不超过10个，找第几个数因为编号不超过10也可以暴力找。
时空复杂度	时间 $O(n \log^2 n)$ ，空间 $O(n)$ 。

题46

试题编号	GCJ 2014 Final D
试题名称	Paradox Sort
题目大意	算法讨论
给定矩阵 a ，满足 $a_{i,j} \neq a_{j,i}$ ，从左到右扫一遍，首先令 t 为第一个元素，并且如果当前元素是 x ，那么若 $f_{x,t}$ 则 t 变为 x ，求字典序最小的排列使得最后留下的数是给定的值 val 。	如果 $a_{i,j} = 1$ 则 i 向 j 连边，那么有解当且仅当 val 能 dfs 访问到所有点。我们从前往后枚举每个数的大小，然后判定可以先 dfs 一遍然后看看没访问到的点能否全部放到当前前缀的 t 后面使得 t 最后留下来即可。注意特判枚举到 val 时的情况。
时空复杂度	时间 $O(T \times n^4)$ ，空间 $O(n^2)$ 。

题47

试题编号	CODEFORCES 331C
试题名称	The Great Julya Calendar
题目大意	算法讨论
给定一个数，每次可以减去它的一个数位，求减少到0的最少步数。	注意到贪心地每次去掉最大数是正确的。用 $dp_{i,j,k}, num_{i,j,k}$ 分别表示 i 位的 $99 \dots k$ ，前面最大数为 j ，要减少到负数的最优答案和答案去掉的数的总和。最后对 n 不断地去掉后面最长的 $99 \dots k$ 即可。
时空复杂度	时间 $O(bit \times 10 \times 10 \times 10)$ ，空间 $O(bit \times 10 \times 10)$ ， bit 表示 n 十进制下的位数。

题48

试题编号	CF 268D
试题名称	Wall Bars

题目大意	算法讨论
求一个长度为 n 的，每一位为1到4的序列，使得存在一种数，相邻位置的差、第一个元素离起点的距离、最后一个元素离终点的距离均不超过 h 。	用 $dp_{n,a,b,c,d}$ 表示前 n 为，三种数距离当前位置的距离为 a, b, c ，且另一种数距离当前位置的距离为1，且若 $d = 1$ 则表示另一种数已经没有连通，在这种情况下的方案数。转移显然。需要卡常数即可。
时空复杂度	时间 $O(nh^3)$ (常数很小)，空间 $O(h^3)$ (滚动数组)。

题49

试题编号	CODEFORCES 286D
试题名称	Tourists
题目大意	算法讨论
数轴有很多个区间，它们出现的时间都不同，询问多次在给定的时间时从原点出发有多长时间是在某个区间中。	在区间相交部分取最早出现的区间，就可以使得区间均不相交。然后用一个前缀和维护即可。
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。

题50

试题编号	CODEFORCES 360D
试题名称	Levko and Sets
题目大意	算法讨论
有两个整数数组 $\{A_1, A_2, \dots, A_n\}$ 、 $\{B_1, B_2, \dots, B_m\}$ 与一个质数 p 。生成 n 的集合，第 i 个集合的生成方式是：一开始这个集合一开始只有一个元素“1”；每次从这个集合中任意选出一个元素 c ，对于所有的 j ，将集合与 $\{(c \times A_i^{B_j}) \bmod p\}$ 取并，直到无法继续增加元素。求这 n 个集合的大小是多少。	求出 $p-1$ 的所有约数，并求出最小的 d_i 使得 $A_i^{d_i} \bmod p = 1$ ，这样的 d_i 一定是 $p-1$ 的约数，可以直接枚举。然后求出 B 中所有元素的 gcd ，再与 $p-1$ 求 gcd ，然后乘入 d_i 中再将 d_i 与 $p-1$ 取 gcd ，可以证明答案就是所有剩余系下是某个 d_i 倍数的数的个数，容斥即可。此外由于CF上的数据很弱，CF上有很多错误代码只能获得75分，请注意。
时空复杂度	时间 $O(m + d(p) \times (n + d(p)))$ ，空间 $O(n + m + d(p))$ ，其中 $d(x)$ 表示 x 的约数个数。

题51

试题编号	Codeforces 316G3
试题名称	Good Substrings
题目大意	算法讨论
给定母串和一些其它串 a_i ，每个 a_i 对应一对 l_i 和 r_i ，问母串中有多少个不同的子串对于所有的 i 它在 a_i 中的出现次数在 l_i 和 r_i 之间。	对于母串和子串中间用一个其它字符连接的串相连并建出这个串的 sam ，在 $parent$ 树中每一个满足它的子树中有对应母串位置的叶子节点且对应任何一个 a_i 中位置的叶子节点的个数在 l_i 和 r_i 之间的节点，答案加上它的对应长度区间的大小即可。
时空复杂度	时间 $O(tot_len)$ ，空间 $O(tot_len)$ ， tot_len 表示母串和所有 a_i 的长度之和。

题52

试题编号	Google Codejam World Final 2014 E	
试题名称	Allergy Testing	
题目大意	算法讨论	
n 种物品中有一个特别的物品，每次可以选一个集合， A 天之后可以知道这个集合内有没有特别的物品，如果有则另需要另一给定天数 $B-A$ 天恢复，最小化最坏情况下的所需天数。	注意到考虑 n 天能分辨出的物品个数，如果能算出则可以二分答案。 n 天情况可以构成一棵二叉树，左边连一条长度为长度为 A 的边，子树为 $n-A$ 天的树，另一侧是长度为 B 的边，子树为 $n-B$ 天的树。考虑叶子往上到根的路径中的不同种组合，若给定 A 的数量和 B 的数量，则每一个组合对应一个组合数，可以直接算出。 B 的数量不多，并且 B 的数量固定时 A 的数量是一个区间，可以用组合数前缀和公式优化。	
时空复杂度	时间 $O(T \times cnt^2 \times \log W)$ ，空间 $O(1)$ ， cnt 是路径上的 B 的个数的最大值，显然是 $O(\log n)$ 的， W 是答案的最大值。	

题53

试题编号	USACO 2012 Dec gold 2	
试题名称	First	
题目大意	算法讨论	
给定一些字符串，您可以改变字母的先后顺序，求可能字典序最小的所有串。	建出trie树然后遍历的时候对所需的字母先后关系连边，如果最后是个DAG图即为可能字典序最小。	
时空复杂度	时间 $O(n \Sigma ^2)$ ，空间 $O(n \Sigma ^2)$ ， Σ 为字符集。	

题54

试题编号	Codeforces 235D	
试题名称	Graph Game	
题目大意	算法讨论	
给定一张 n 个点 n 条边的图，求随机点分治的期望复杂度。	随机点分治即随机给点分配权值，并且对于一对点 (u, v) 若存在 u 到 v 的一条路径使得 u 的权值是该路径上最大的则答案贡献一，那么我们要求对 (u, v) 存在这样一条路径的概率是多少，若它们之间只有一条路径则直接答案加上 $\frac{1}{L}$ ，否则利用容斥原理即可。	
时空复杂度	时间 $O(n^2)$ ，空间 $O(n^2)$ 。	

题55

试题编号	GCJ 2009 Final B	
试题名称	Min Perimeter	
题目大意	算法讨论	
给定平面上的一些点，求三个点组成的三角形的最小周长。	利用类似求最近点对的分治算法，以及利用三角形两边之和大于第三边的性质，容易设计出算法。	
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。	

题56

试题编号	codeforces - 267C
试题名称	Berland Traffic
题目大意	算法讨论
给一张网络和源汇，求最大流并且满足对所有点所有源到它的路径上的流量之和都是相等的。	用 x_i 表示起点到点 i 的流量，容易对每个点利用流量平衡条件列出方程，加上 $x_S = 1$ ，并强制 $x_T = 1$ ，解之即可，注意特殊情况的处理。算流量总和可以通过计算起点出发的所有边的流量和。
时空复杂度	时间 $O(n^3)$ ，空间 $O(n^2)$ 。

题57

试题编号	Codeforces 261D
试题名称	Maxim and Increasing Subsequence
题目大意	算法讨论
多次给定一个序列，长度为 n ，序列的数不超过 m ，求该序列复制 t 遍后的最长上升子序列。	t 是不会超过 $\min\{n, m\}$ 的。不然能直接输出。然后记录 dp_i 表示当前末尾为 i 的答案，循环整个序列 t 次，每次暴力从当前位置的数开始从小往大暴力更新 dp ，由于每个 dp 值最多增加 $\min\{n, m\}$ 次，这样做时间复杂度可以保证。
时空复杂度	时间 $O(nm)$ ，空间 $O(n + m)$ 。

题58

试题编号	Codeforces 329E
试题名称	Evil
题目大意	算法讨论
给定平面上一些点，求最小曼哈顿距离哈密顿回路。	取 x 坐标和 y 坐标的中位数 x_m, y_m ，将 $2 \sum_{k=0}^{k \leq n} (x_k - x_m + y_k - y_m)$ 。
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。

题59

试题编号	codeforces 253E
试题名称	Printer
题目大意	算法讨论
有 n 个任务，每个任务有个完成时间、出现时间及优先级，每个时刻会完成当前优先级最高的任务，有一个任务优先级未确定但完成时间确定。求这个任务的优先级以及所有任务的完成时间。	给定优先级模拟过程计算完成时间可以通过优先队列解决，求优先级可以利用优先级和完成时间的单调性二分。
时空复杂度	时间 $O(n \log^2 n)$ ，空间 $O(n)$ 。

题60

试题编号	codeforces 306D
试题名称	Polygon

题目大意	算法讨论
Polycarpus喜欢凸多边形，尤其喜欢每个角角度都相同，每条边长度都不相同的凸多边形。你需要在知晓给定的顶点数的情况下，为他描绘出这样的任意一个凸多边形。	如果 $n \leq 4$ ，显然无解。先计算出每个角的角度 θ ，令第一条边是 x 轴正方向上的一条边，根据向量旋转公式每次旋转 θ 角。对于前 $n-1$ 条边，直接令第 k 条边的长度分别为 $1 + 0.001k$ 。对于第 n 条边算出它与 x 轴的交点，也几乎不可能与前面长度有重合的。
时空复杂度	时间 $O(n)$ ，空间 $O(1)$ 。

题61

试题编号	codeforces 341E
试题名称	Candies Game
题目大意	算法讨论
给定 n 个盒子，每个盒子里有一定数目的糖果，最多不超过 a 个，现在每次可以将一个盒子中的糖果全部转移到另一个中，求一个方案使得最后只有两个盒子里有糖果。	给定三个非零数 $a \leq b \leq c$ ，设 $b = ax + y$ ，通过二进制表示 x ，并且利用 c 不小于 b 的性质利用 c 做过渡，很容易实现将 b 转化为 $b \bmod a$ 的过程，显然通过这样的过程很容易完成任务。
时空复杂度	时间 $O(n \log^2 a)$ ，空间 $O(n)$ 。

题62

试题编号	codeforces 346E
试题名称	Doodle Jump
题目大意	算法讨论
数轴上有 n 个点，第 x 个点的坐标是 $ax \bmod p$ ，其中 $(a, p) = 1$ 。问从原点出发，每次能跳不超过 h 的距离，能否跳到最大的那个点。 T 次询问。	即询问最大间隔。按 x 顺序依次产生点，注意到 x 扫完整个数轴一遍之后，留下了很多段，第二次相当于是给每一段的某个位置都加一个点，第三边是另一个位置，且每次加的位置构成一个子问题，可以递归计算。
时空复杂度	时间 $O(T \log a)$ ，空间 $O(\log a)$ 。

题63

试题编号	codeforces 325E
试题名称	The Red Button
题目大意	算法讨论
给定 n ，求构造一个 $0 \leq a_i < n$ 的排列 a ，满足在模 n 意义下， a_i 要么等于 $2a_{i-1} + 1$ 要么等于 $2a_{i-1}$ 。且 $a_{n-1} = 0$ 或者 $2a_{n-1} + 1 = 0$ 。	n 为奇数显然无解。令 $a_n = 0$ ，若当前得到了 a_i ，若可以填 $\lfloor \frac{a_i + n}{2} \rfloor$ 则填之，否则可以证明 $\lfloor \frac{a_i}{2} \rfloor$ 。
时空复杂度	时间 $O(n)$ ，空间 $O(n)$ 。

题64

试题编号	codeforces 241E
试题名称	Flights

题目大意	算法讨论
现有一个 DAG ，每条边的边权都是1。要求你把某些边的边权改成2，使得任何一条1到 n 的路径的长度相等。	设起点到点 i 的路径长度为 x_i ，可以列出一堆的不等式，加上 $x_0 = 0$ ，直接上差分约束系统就行了。
时空复杂度	时间 $O(SPFA(n, m))$ ，空间 $O(n + m)$ 。

题65

试题编号	codeforces 241E
试题名称	Flights
题目大意	算法讨论
现有一个 DAG ，每条边的边权都是1。要求你把某些边的边权改成2，使得任何一条1到 n 的路径的长度相等。	设起点到点 i 的路径长度为 x_i ，可以列出一堆的不等式，加上 $x_0 = 0$ ，直接上差分约束系统就行了。
时空复杂度	时间 $O(SPFA(n, m))$ ，空间 $O(n + m)$ 。

题66

试题编号	USACO March Contest 2009, Gold Division, Problem 2
试题名称	Cleaning Up
题目大意	算法讨论
给定一个序列，可以分成很多段，每一段的权值为这一段不同的数的个数的平方，求最小权值。	显然每一段的不同的数不能超过 \sqrt{n} ，所以用 dp_i 表示以第 i 个数结尾的答案，然后扫一遍序列同时单调地维护最前的满足它到当前位置的区间中的不同的数不超过 $1, 2, 3, \dots, \sqrt{n}$ 即可。
时空复杂度	时间 $O(n\sqrt{n})$ ，空间 $O(n\sqrt{n})$ 。

题67

试题编号	CODEFORCES 316D
试题名称	PE lesson
题目大意	算法讨论
给定一个排列，可以交换一些位置，每个位置 i 最多交换 b_i 次且 $b_i \leq 2$ 。求最终可能的方案数。	给定一个置换，将其拆为轮换，则可以证明每个轮换里的 b 数组为1的个数不能多于2。枚举有几个轮换里只有1个数的 b 值等于1，由此算出有几个2个数的 b 值等于1，然后用组合数学方法计算答案即可。
时空复杂度	时间 $O(n\sqrt{n})$ ，空间 $O(n\sqrt{n})$ 。

题68

试题编号	Codeforces 273D
试题名称	Dima and Figure

题目大意	算法讨论
求 $n \times m$ 的矩阵中的所有联通块，满足任意两点可以经过联通块内的点可以在它们的曼哈顿距离之内到达。	恶心题，用 $dp[i][j][k][a][b]$ 表示第 i 列，上面是第 j 行，下面是第 k 行，且现在上边界的上升情况为 a ，下边界的上升情况为 b 的答案，然后转移，并用二维前缀和优化。
时空复杂度	时间 $O(n^3)$ ，空间 $O(n^3)$ 。

题69

试题编号	Codeforces 311E
试题名称	Biologist
题目大意	算法讨论
有 n 个物品，每个物品有一个属性 $a_i = 0, 1$ ，且可以用 p_i 的代价改变其属性。有 m 个条件集合，第 k 个条件集合大小为 sz_k ，集合中每个条件都是要求某个物品的属性为某个值，如果全部满足了就可以得到一定的收益，有些条件集合不满足会有一定的代价，求最大收益。	最小割裸题，直接建模即可。
时空复杂度	时间 $O(MaxFlow(n + m, n + \sum sz_k))$ ，空间 $O(n + m + \sum sz_k)$ 。

题70

试题编号	Codeforces 251D
试题名称	Two Sets
题目大意	算法讨论
有一个数列，其中的数不超过 a ，求划分成两个集合 A, B ，它们中的元素的异或和分别为 sum_a, sum_b ，求最大化 $sum_a + sum_b$ ，同时最小化 sum_a 。	将整个集合的异或和求出来并将和为1的位置去掉，然后高斯消元求出线性空间的基，之后由高位往低位贪心即可。
时空复杂度	时间 $O(n \log a)$ ，空间 $O(n)$ 。

题71

试题编号	Codeforces 332D
试题名称	Theft of Blueprints
题目大意	算法讨论
n 个点的带权无向图，点 i 与点 j 之间有一条权值为 $c_{i,j}$ 的边。对任意大小为 k 的顶点集合 S ，恰有一个点与 S 的每个点都有边，这个集合代价为该点所连所有边边权之和，求任选大小为 k 的集合的期望代价。	对一个度数为 deg 的点，对总集合数有 $\binom{deg}{k}$ 的贡献，对于边 $u \rightarrow v$ ， u 的度数为 deg ，那么它会出现 $\binom{deg-1}{k-1}$ 个集合中。据此很容易计算答案。
时空复杂度	时间 $O(n^2)$ ，空间 $O(n^2)$ 。

题72

试题编号	Codeforces 332D
试题名称	Theft of Blueprints

题目大意	算法讨论
给定图的一部分，要求有多少个不同的图是满足这些条件的：从点 i 出发，可以到达点 $i+1, i+2, \dots, n$ ；任意从 u 到 v 的有向边满足不等式： $u < v$ 。两点之间最多有一条边。对于一对点 $i, j (i < j)$ ，若 $j-i \leq k$ ，那么从 i 到 j 的最短距离等于 $j-i$ 。对于一对点 $i, j (i < j)$ ，若 $j-i > k$ ，那么从 i 到 j 的最短距离等于 $j-i$ 或 $j-i-k$ 。	易知点 i 必须向 $i+1$ 连边，可以向 $i+k+1$ 连边。且若两个点 i, j 均连了后者，则 $i+k+1 > j$ 。然后用一些组合数学的知识计算即可，注意特殊情况的处理。
时空复杂度	时间 $O(n)$ ，空间 $O(n)$ 。

题73

试题编号	Codeforces 348E
试题名称	Pilgrims
题目大意	算法讨论
有一棵黑白树，你可以摧毁一个白点，一个黑点不高兴当且仅当他不能到达任何在原树中离他最远的黑点。要你最大化不高兴的黑点数目以及求出方案数。	求出树的直径，注意到每个点到最远点的路径一定与该直径有交点，记录一些东西并分些情况讨论即可。
时空复杂度	时间 $O(n)$ ，空间 $O(n)$ 。

题74

试题编号	Codeforces 293B
试题名称	Distinct Paths
题目大意	算法讨论
有一个 $n \times m$ 的木板，一些块已经被涂上给出的 k 种颜色中的一种。你需要把每个没涂色的块涂色使得从左上角到右下角的每条路径都不会经过两个颜色一样的块。路径只能向右或向下走。	易知若 $n+m-1 > K$ 则一定无解，否则 n, m 均很小，可以用搜索剪枝或者优化的状压DP解决。
时空复杂度	若使用状压DP则时间 $O(nmS)$ ，空间 $O(nm+S)$ ， S 是状压DP的状态数，是比较小的。

题75

试题编号	USACO 2005 December Gold
试题名称	Cow Patterns
题目大意	算法讨论
给定一个母串 S 和一个模式串 s ，求所有的母串中，长度与模式串相同且任意两个位置之间的大小关系($>, <, =$)均与模式串对应位置相同的所有子串。	对每个位置按(前面比该位置小的有多少个，前面与这个位置相同的有多少个)建立hash函数。然后维护即可。
时空复杂度	时间 $O((S + s) \Sigma)$ ，空间 $O((S + s) \Sigma)$ ， S 是状压DP的状态数，是比较小的， Σ 为字符集。

题76

试题编号	Codeforces 258D
试题名称	Little Elephant and Broken Sorting
题目大意	算法讨论
给定一个排列，有 m 个交换，每个交换有50%的概率不操作，求最后期望的逆序对数。	设 $dp_{i,j}$ 为第 i 个位置比第 j 个位置大的概率，很容易实现上述操作。
时空复杂度	时间 $O(n(n+m))$ ，空间 $O(n^2)$ 。

题77

试题编号	Codeforces 354D
试题名称	Transferring Pyramid
题目大意	算法讨论
一个 n 层的三角形平面金字塔，你能进行两种操作。第一个操作是给某个点染色，代价是3。第二个是给一个子三角形染色，底边必须是金字塔的底边，代价是该三角形的点数+2。现在有 m 个黑点。要你覆盖所有黑点。	容易设计出暴力DP方程，然后容易发现，只有最后 $O(\sqrt{n})$ 行才有可能有第二个操作，由此可以优化复杂度。
时空复杂度	时间空间均为 $O(n\sqrt{n})$ 。

题78

试题编号	Codeforces 333C
试题名称	Lucky Tickets
题目大意	算法讨论
如果一个允许前导0的八位数满足在这些数之间插入运算符和括号能得出一个 k ，那么这个数是 k 幸运数。求 m 个 k 幸运数。	随机化前四位数的数和运算符，算出后四位数应该加上或者减去多少，并用 set 或 $hash$ 判重，很容易找出 m 个 k 不同的幸运数。
时空复杂度	时间 $O(10^8)$ ，空间 $O(10^8+m)$ 。

题79

试题编号	Codeforces #185
试题名称	Fetch the Treasure
题目大意	算法讨论
有一个数轴，和一个集合 S ， m 次操作，每次可以从集合里选一个数 x ，并往前跳 x 步，一开始在原点，数轴上 n 个位置有一个权值，一开始集合里有一个数 K ，每次可以往集合里加入不超过 K 的数，或者减少一个位置的权值，或者询问当前能到的最大权值的点权值为多少。	加入数的次数很少，我们考虑对模 k 意义下的每一个位置，要到达这个位置，至少要走过几个 k ，也就是对 $a(a < k)$ ，求出最小的 $b \in N$ 使得 $bk+a$ 能到达，这个用 dij 很容易解决，加入一次数就做一次最短路，然后优先队列维护即可。
时空复杂度	时间 $O(m \log n + t^2 n \log n)$ ，空间 $O(tn)$ ， t 为向集合中加入数的操作次数。

题80

试题编号	Codeforces 285E
试题名称	Positions in Permutations

题目大意	算法讨论
P是n个互不相同且不超过n的正整数的一个排列。我们设排列P的第i个元素为 P_i ，n为排列的长度。我们称排列中的第i个位置是完美的，当且仅当 $ P_i - i = 1$ 。请求出长度为n的而且完美的位置数刚好为k的排列数是多少。答案要求取模 $10^9 + 7$ 。	用 $dp[i][j][0/1][0/1]$ 表示当前到了第i个数，前面至少有j个完美，第 $i - 1$ 个数是否等于 $i - 1$ ，第i个数是否等于i的，只考虑所有完美位置的方案数。 利用容斥原理和组合计算很容易求得最终答案。
时空复杂度	时间 $O(n^2)$ ，空间 $O(n^2)$ 。

题81

试题编号	Codeforces 332E
试题名称	Binary Key
题目大意	算法讨论
给定一个母串 p ，和模式串，求字典序最小的长度为k的01钥匙，使得它不断重复写直到和母串长度相等时，取出母串所有对应1的位置之后形成的串恰好是模式串。	枚举模式串1的个数，可以先排除掉一些不可能的答案加速，然后从后往前枚举能否填1即可。
时空复杂度	时间 $O(k p)$ ，空间 $O(p)$ 。

题82

试题编号	Codeforces 293D
试题名称	Ksusha and Square
题目大意	算法讨论
给定一个凸多边形，求从中选两个格点作为对角线的正方形的面积的期望。	正方形的面积即连线长度的平方除以二，可以表示成坐标平方和和坐标乘积和，通过扫描 x 坐标，可以维护坐标平方和和坐标和，于是容易计算出坐标乘积和。
时空复杂度	时间 $O(n + 10^6)$ ，空间 $O(n + 10^6)$ 。

题83

试题编号	Codeforces 249E
试题名称	Endless Matrix
题目大意	算法讨论
有一个矩阵，里面有连续的数字，从1开始。若 $a_{i,j} < a_{t,k}$ ($i, j, t, k \geq 1$)当且仅当 $\max(i, j) < \max(t, k)$ 或 $\max(i, j) = \max(t, k)$ 且 $j < k$ 或 $\max(i, j) = \max(t, k)$ 且 $j = k$ 且 $i < t$ 。	可以用容斥原理，转化为一个(1, 1)为左上角的矩形的权值和。不妨设 $n \leq m$ ，那么左上角 $n \times n$ 的矩阵是一个平方和，右边会多出一个矩形，每一列是等差数列，每一行是差后等差数列，容易推导出公式计算。
时空复杂度	时间 $O(T)$ ，空间 $O(1)$ 。

题84

试题编号	Codeforces 325C
试题名称	Monsters and Diamonds

题目大意	算法讨论
有 n 种怪物，每种怪物都有一个 $1 \rightarrow n$ 的唯一编号。现在有 m 个规则，每个规则是将一个怪物分裂成若干个怪物与若干个钻石。求对每个怪物，解决完所有的怪物之后，能得到的钻石的最小值和最大值。	最小值显然可以用类 $dijkstra$ 算法解决，最大值可以考虑用记忆化搜索的方式来 dp 。有环显然就是无穷大，根据最小值我们能够轻松分辨出一个点能否分裂完。然后在保证只会走到能分裂完的点的前提下，记忆化搜索，对于环就直接设为无穷大，就能解决最大值问题。
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。

题85

试题编号	Codeforces 269E
试题名称	String Theory
题目大意	算法讨论
一架长方形的竖琴是由 n 行 m 列构成的网格。在竖琴的左右两侧各有 n 个钉子，上下边界各有 m 个钉子。同时，这架竖琴恰好有 $n + m$ 根不同的琴弦，每一根琴弦的两端分别被钉子固定在竖琴不同侧的边界上，而且每个钉子上有且仅有一根琴弦。如果存在两根互相交叉的琴弦，则这架竖琴将无法弹奏。为了使得这架竖琴可以被用来弹奏，可以选择不同的两列，对应地交换处于同侧的钉子；或者选择不同的两行，对应地交换处于同侧的钉子，两侧必须同时交换且不改变钉子与其所固定的琴弦的连接。找出对于初始时的每行每列，在最后的竖琴上所应该处在的位置。	我们从任意点出发，按走到其所在琴弦的另一端，然后走到该点对面的点，再继续走到琴弦的另一端，反复这个过程，类似置换拆为轮换，最后显然会形成一个环。如此会将一个竖琴看做若干个环。我们把一个环上每个点所在的边界类型写下来形成字符串。我们对于唯一可能的目标状态，也求出来这样的很多字符串，并且如果两个字符串能旋转、翻转同构那么两者可以匹配，如果能将初态与终态的字符串一一匹配显然有解，且可以求出来，通过最小表示法很容易实现。
时空复杂度	时间 $O(n \log n)$ ，空间 $O(n)$ 。

题86

试题编号	Codeforces 323C
试题名称	Two permutations
题目大意	算法讨论
你有两个各包含 n 个元素的排列 p 和 q ，和 m 个由 l_1, r_1, l_2, r_2 组成的询问。每次询问在 p 中位置在 $[l_1, r_1]$ ，在 q 中位置在 $[l_2, r_2]$ 中的数的数量。一个 n 元素的排列是指 n 个不同的数，每个数在 $1-n$ 之间。数字 v 的位置在排列 g_1, g_2, \dots, g_n 中是使得 $g_i = v$ 的 i 。强制在线。	容易将问题转化为二维数点问题，用主席树维护前缀线段树即可。
时空复杂度	时间 $O((n + m) \log n)$ ，空间 $O(n \log n)$ 。

题87

试题编号	GCJ 2009 Final D
试题名称	Wi-fi Towers

题目大意	算法讨论
给定一些点，每个点有一个半径和一个有正有负的权值。求权值和最大的点集，使得若A在集合中则距离A不超过其半径的点也在集合中。	由于数据水直接最大权闭合子图即可。
时空复杂度	时间 $O((n+m)\log n)$ ，空间 $O(n\log n)$ 。

题88

试题编号	Codeforces, MemSQL start[c]up Round 2 - online version
试题名称	Counting Skyscrapers
题目大意	算法讨论
大街上建好了一排摩天大楼。摩天大楼的数量是在2到314!中均匀随机选择的。每座摩天大楼的楼层数是被独立地随机选择的：对于每个正整数 i ，楼层数为 i 的概率为 2^{-i} 。如果一座摩天大楼有 i 层，那么它的楼层被编号为0到 $i-1$ 。摩天大楼间修建了一些滑索。一座摩天大楼的第 i 层和另一座摩天大楼的第 j 层之间有滑索当且仅当两楼之间没有摩天大楼有第 i 层。Alice是个严谨认真的人，测完后她的计数器是摩天大楼的数目的准确值。Bob把计数器初始化为1，从最左边的摩天大楼开始利用滑索从一座摩天大楼滑到另一座。然后在高度不大于给定值 h 的往右滑的滑索中选一个最高的。若楼层为 i ，他直接将计数器加上 2^i 。他会一直持续这个过程直到他到达最右边的摩天大楼。当Alice和Bob到达最右边的摩天大楼时，他们会比较计数器的值。现在给出Alice的计数器的值或者Bob的计数器的值，请你求出另一人的计数器的值的期望。	314!可以看做inf。 Alice的答案为 n 。 Bob的答案为 $\sum_{l=1}^{n-1} (n-l)2^{-h}(1-2^{-h})^{l-1}\left(\frac{1}{2} - \frac{1}{2} \times \frac{l-1}{2^h-1}\right)$
时空复杂度	时间 $O(nh)$ ，空间 $O(1)$ 。

题89

试题编号	Codeforces 254D
试题名称	Rats
题目大意	算法讨论
一个地图，里面有一些老鼠。为了清除老鼠，某人选择丢两个手榴弹。地图是 $n \times m$ 的网格图，里面有些位置有墙，其余位置是空的。有些位置有老鼠，老鼠们睡着了不会动。你只能在没有墙地方放炸弹，每秒钟，炸弹的波及范围会延伸一格，但是炸弹的延伸过程是不会波及到墙的，炸弹在爆炸 d 秒后会瞬间消失。问是否能找到两个格子放置炸弹，使得炸死所有老鼠。	炸死第一只老鼠的炸弹的位置一定是到它的最短路不超过 d 的点集，不超过 d^2 个。暴力枚举之，然后找到一个没被炸死的老鼠，在它附近的 $O(d^2)$ 个位置枚举第二个炸弹。之后判定是否炸死所有老鼠即可。

时空复杂度	时间 $O(d^6 + nm)$, 空间 $O(nm)$ 。
-------	---------------------------------

题90

试题编号	Codeforces 243D	
试题名称	Cubes	
题目大意		算法讨论
有一个 $n \times n$ 的网格, 第 i 行第 j 列上面放有 $a_{i,j}$ 个立方体。现在有无数方向向量为 $(s_x, s_y, 0)$ 的平行光束从无穷远处射来, 求它能射到的立方体数目。		旋转坐标使向量方向与x轴平行, 然后转化为简单的区间操作问题, 用线段树即可。
时空复杂度	时间 $O(n^2 \log n)$, 空间 $O(n^2)$ 。	

题91

试题编号	Codeforces 283E	
试题名称	Cow Tennis Tournament	
题目大意		算法讨论
有 n 头奶牛, 每两个奶牛有个武力值 a_i , 它们两两间会有比赛。武力值互不相同, 两头奶牛比赛的胜负性初值给定。有 m 个操作, 每次操作为把一段区间里任意两头牛的胜负关系取反。最后要你求有多少三元组 (a, b, c) 满足 a 胜 b , b 胜 c , c 胜 a 。		求第 k 个点赢得场数为 d_i , 那么 $\binom{n}{3} - \sum_{i=0}^{i \leq n} d_i(d_i - 1)/2$ 就是答案。 排序后重编号。改变关系很容易用扫描线离线解决。
时空复杂度	时间 $O(n \log n)$, 空间 $O(n)$ 。	

题92

试题编号	Codeforces 263E	
试题名称	Rhombus	
题目大意		算法讨论
给定一张 $n \times m$ 的表, 第 i 行第 j 列的数为 $a_{i,j}$, 求整数对 (a, b) 满足 $k \leq a \leq n - k + 1, k \leq b \leq m - k + 1$, 设 $f(x, y) = \sum \sum a_{i,j} \times \max\{0, k - i - x - j - y \}$, 求 $f(a, b)$ 的最大值。		旋转变换后, 通过计算前缀和并利用容斥原理, 容易解决该题。实际上由于数据较弱, 直接暴力也可以通过。本题还可以用维护二阶偏差分的方法做。
时空复杂度	时间 $O(n^2)$, 空间 $O(n^2)$ 。	

题93

试题编号	Codeforces 301E	
试题名称	Yaroslav and Arrangements	

题目大意	算法讨论
定义一个数列 a_1, a_2, \dots, a_r 为良好序列, 当且仅当: $ a_1 ? a_2 = a_2 ? a_3 = \dots = a_{r-1} ? a_r = a_r ? a_1 = 1$, a_1 是数列中的最小值。定义一个数列 b_1, b_2, \dots, b_r 为优秀序列, 当且仅当: $b_i \leq b_{i+1} (1 \leq i < r)$ $1 \leq r \leq n, 1 \leq b_i \leq m$, 且通过重排可得到至少一个至多 p 个良好序列, 给定 n, m, p , 求优秀序列的数目。	我们用(最大值-最小值, 最大值个数 x , 总元素数, 能转化为的良好数列数)。若 x 表示当前最大值, 转移就是枚举放几个 $x+1$, 然后用简单的组合计算解决问题。时间复杂度看起来是 $O(n^5)$ 的, 不能忍受, 但实际上有效状态量没有这么多。只算有效状态量的来写, 并用滚动数组即可。
时空复杂度	时间 $O(n^5)$ 但实际达不到, 空间 $O(n^3)$ 。

题94

试题编号	Codeforces 325D
试题名称	Reclamation
题目大意	算法讨论
一个 $r \times c$ 的四联通左右形成环形的地图。每次要删除一个点, 并查询是否有从上边界走到下边界的一条路, 如果没有则恢复上一次删除。执行多次问删除点数。	如果非环形转对偶图, 即将障碍转为点, 将点转为障碍, 化为八连通图, 来查询是否能从左边找一条路径到右边即可完成上述操作。环形的话可以证明环翻倍即可。直接用并查集即可解决问题。
时空复杂度	时间 $O(rca(n))$, 空间 $O(rc)$ 。

题95

试题编号	Codeforces 274C
试题名称	The Last Hole
题目大意	算法讨论
有 n 个点, 每个点向外扩张成圆, 每个点的扩张速度相等。求最早的一个时刻, 使得之后再也没有圆中的“洞”。	只有点所构成的锐角三角形的外心以及矩形的中心才有可能会是答案。容易做到 $O(n^3)$ 枚举 $O(n)$ 更新答案。
时空复杂度	时间 $O(n^4)$, 空间 $O(n)$ 。

题96

试题编号	USACO Open Contest 2008
试题名称	Cow Neighborhoods
题目大意	算法讨论
有 n 个点, 两点有边当且仅当其曼哈顿距离不超过一个定值 R , 求连通块个数及点数最多的连通块有几个点。	先将点 (x, y) , 变为 $(x+y, x-y)$, 由此可将一个点所能影响到的区域由菱形变成正方形。按横坐标排序, 然后考虑对某个点所有横坐标比他小且与它横坐标距离不超过 R 的点, 将该点 y 坐标上的前驱和后继能连边则连边即可。
时空复杂度	时间 $O(n \log n)$, 空间 $O(n)$ 。

题97

试题编号	Codeforces 260E
试题名称	Dividing Kingdom
题目大意	算法讨论
平面上有个 n 个点以及一个长度为9的数列 a ，满足和为 n 。请在竖直和水平方向上分别划两条线，将平面分成9个部分且这9各部分的点数经过重排列可以变成 a 。	枚举 a 的排列，然后根据左上角的四个方块可以确定四条线，然后记录几个前缀和完成总的判定即可。
时空复杂度	时间 $O(n + 9!)$ ，空间 $O(n)$ 。

题98

试题编号	Codeforces 248E
试题名称	Piglet's Birthday
题目大意	算法讨论
有 n 个架子，一开始每个架子有 a_i 个满的蜜罐。有 m 次操作 u_i, v_i, s_i ，表示从 u_i 个架子上随机拿 s_i 个蜜罐，并把这些蜜罐里的蜜都喝光，然后放到 v_i 上，并每次回答所有蜜罐都空了的架子数目的期望值。	用 $P_{i,j}$ 表示第 i 个架子上有 j 个蜜罐的概率，显然答案为 $\sum P_{i,0}$ 。考虑维护。记函数 $F(n, m, k, l)$ 为 n 个物品，有 m 个特殊物品，随机选 k 个，正好选中 l 个特殊物品的概率。有了 F 函数，转移很容易实现。 F 函数本身也容易用组合数计算。
时空复杂度	时间 $O(mas)$ ，空间 $O(na)$ 。

题99

试题编号	Codeforces 243C
试题名称	Colorado Potato Beetle
题目大意	算法讨论
有一个长宽均为 $10^4 + 1$ 的田地。上面画了 n 条线段，线段不会出地图外面。所有被线段经过的格子都会撒上杀虫剂。虫子从最外层开始入侵，且无法走到有杀虫剂的格子。求最后没有被破坏的格子数目。	离散化后，我们在新图进行一遍floodfill，然后直接找一个最外面的点开始模拟，并统计没有经过的点数。
时空复杂度	时间 $O(n^2)$ ，空间 $O(n^2)$ 。

题100

试题编号	Codeforces 266E
试题名称	BerDonalds
题目大意	算法讨论
一个 n 个点 m 条边无向图，找一个点(可能在某条边上)，最小化离它最远的点到它的距离。输出该距离。	求出最小直径生成树后，最小直径除以2即为答案，最小直径生成树的求法很容易找到，这里不再赘述。
时空复杂度	时间 $O(n^3)$ ，空间 $O(n^2)$ 。