

## CodeChef 题目泛做

石家庄二中 张若天

2016 年 1 月 22 日

试题编号	Codechef Nov 14
名称	Chef and Churu
题目大意	一个 $N$ 个数字的数组 $A$ ，元素标号 $1$ 到 $N$ ，同时有标号为 $1$ 到 $N$ 的 $N$ 个函数。第 $i$ 个函数会返回数组中 $L[i]$ 到 $R[i]$ 之间的元素和。支持修改数组某个元素的值。询问标号在 $m$ 到 $n$ 之间函数值的和。 $1 \leq N \leq 10^5, 1 \leq A_i, y \leq 10^9, 1 \leq Q \leq 10^5$ 。
算法讨论	对函数按照编号分块，预处理出数组中每个元素在每一块中的出现次数，每块维护函数和。再用树状数组维护数组的区间和。对于修改可以每块根据出现次数进行修改和树状数组上直接修改。查询两种方式一块查询即可。
时空复杂度	时间： $O(N * \sqrt{N * \log(N)})$ 空间： $O(N * \sqrt{N * \log(N)})$

试题编号	Codechef Nov 14
名称	The Spelling Problem
题目大意	给一个字典和一个可能有 4% 单词错误的小写英文句子，输出更正后的句子。错误一共有 4 种情况。1. 交换了两个字母。2. 漏掉了一个字母。3. 多打了一个字母。4. 打错了一个字母。最多一个单词里只会出现一种错误。文本大小大约是 10MB。
算法讨论	对于字典中每个单词都记录一个权值。代表这个词出现的可能性。首先扫一遍整个串，若出现一次字典中某个词，这个词的权值加一个数。再扫一遍，对于每个不在字典中的词，去尝试每种错误，查询是否为正确单词，若有多个候选正确单词取权值较大的。可以用 hash 和 hash 表实现。
时空复杂度	时间： $O(L + 4\% * L * 26 * 26)$ , 空间： $O(L)$ , $L$ 为句子长度

试题编号	Codechef Nov 14
名称	Sereja and Order
题目大意	N 个任务，每个任务都要在两台机器上运行，第 i 个任务需要在第一台机器上运行 $A_i$ 秒，在第二台机器上运行 $B_i$ 秒。一台机器不能同时运行两个任务，一个任务也不能同时在两台机器上运行。求完成所有任务所需的最少时间。 $1 \leq A_i, B_i \leq 10^5, 1 \leq N \leq 10^4$ 。
算法讨论	可以简单的分析出时间的下界是 $\max(\max_i(A_i + B_i), \max(\sum A_i, \sum B_i))$ 。当 $\max_i(A_i + B_i)$ 取到 max 时，设 $i=p$ ，可以证明 $B_p \geq \sum A_i - A_p$ 和 $A_p \geq \sum B_i - B_p$ 。即可构造一种可行解。当右半边取到 max 时，可以按照 $A_i - B_i$ 的顺序从小到大排序，然后 A 机器正着执行，B 机器反着执行，若某个任务同时在两个机器执行，可以挪到某个机器的最前面（后面）。排序后可以保证 $A_1 + A_2 + \dots + A_i \leq t - B_i$ ，所以交换后就是一组可行解。
时空复杂度	时间： $O(N * \log(N))$ 空间： $O(N)$

试题编号	Codechef Jul 12
名称	Dynamic GCD
题目大意	给一颗 n 个节点的树，第 i 个节点上有正整数权值 $v_i$ ，维护两种操作：链上所有的点的 $v_i$ 加一个数 d，查询链上所有点的 $\gcd(v_i)$ 。 $1 \leq n, Q \leq 5 * 10^4, 0 \leq d \leq 10^4$ 。
算法讨论	树链剖分，每个重链利用线段树维护 gcd。因为 $\gcd(x, y) = \gcd(x, x - y)$ ，所以维护一段数的 gcd 可以维护这段数的差分序列的 gcd。原序列的区间加变成差分序列的两个单点修改，可以方便维护。查询时按照树链剖分的方法查询即可。
时空复杂度	时间： $O(n * \log(n)^3)$ 空间： $O(n)$

试题编号	Codechef Jun 14
名称	Two Companies
题目大意	给一颗 $n$ 个点的树。给你两种颜色的链，分别有 $m_1$ 、 $m_2$ 条，每条链有权值 $v_i$ 。选出一些链使得选出的权值和最大，要求颜色不同的链在树上不能相交。 $1 \leq N \leq 10^5, 1 \leq m_1, m_2 \leq 700, 1 \leq v_i \leq 10^6$ 。
算法讨论	对所有颜色不同并且在树上相交的链连边，就可以把问题转化成一个二分图最大带权独立集问题。可以用网络流的最小割算法求解。
时空复杂度	时间： $O((m_1 + m_2) * \log(n) + \text{Maxflow}(m_1 + m_2, m_1 * m_2))$ 空间： $O(m_1 * m_2 + n * \log(n))$

试题编号	Codechef Feb 13
名称	Observing the Tree
题目大意	给定一棵 $n$ 个节点的树，每个点初始权值为 0。有 $M$ 个操作。第一种操作是将 $X$ 到 $Y$ 链上的点加上一个首项为 $A$ 公差为 $B$ 的等差数列。第二种操作是询问 $X$ 到 $Y$ 链上的权值和。第三个操作是回滚到第 $X$ 次修改后的状态。 $1 \leq N, M \leq 10^5, 0 \leq A, B \leq 1000, 0 \leq X_1, Y_1 \leq 10^5$ 。
算法讨论	可以把整棵树链剖分，然后每个重链用线段树维护。现在需要线段树上支持加等差数列，发现两个等差数列的首项 $A$ 和公差 $B$ 是可以方便合并的，于是可以维护。对于求和操作可以套用等差数列求和公式。回滚操作把整个线段树换为可持久化线段树即可。
时空复杂度	时间： $O(n * \log^2(n))$ 空间： $O((M + N) * \log(n))$

试题编号	Codechef Jul 14
名称	Game of Numbers
题目大意	<p>有两个长度为 <math>N</math> 的数组 <math>A</math> 和 <math>B</math>。维护两个集合 <math>S_1, S_2</math>，初始时均为空。每次操作，他将会选择两个数对 <math>(i,j),(p,q)</math>，满足 <math>(i,j)</math> 不在 <math>S_1</math> 中，<math>(p,q)</math> 不在 <math>S_2</math> 中，<math>B_j &gt; A_i, B_p &lt; A_q, \gcd(A_i, B_j) \neq 1, \gcd(A_q, B_p) \neq 1</math>，且 <math>\gcd(\gcd(A_q, B_p), \gcd(A_i, B_j)) \neq 1</math>。如果这样的数对存在，他会将 <math>(i,j),(p,q)</math> 分别加入到集合 <math>S_1, S_2</math> 中。求最多多少次操作。<math>1 \leq N \leq 400, 1 \leq T \leq 10, 1 \leq A_i, B_i \leq 10^9</math>。</p>
算法讨论	<p>对于 <math>A_i &lt; B_j</math> 且 <math>\gcd(A_i, B_j) \neq 1</math> 的 <math>(i,j)</math> 把 <math>\gcd(A_i, B_j)</math> 加到一个数组 <math>U</math> 里。同样，对于 <math>A_i &gt; B_j</math> 且 <math>\gcd(A_i, B_j) \neq 1</math> 的 <math>(i,j)</math> 把 <math>\gcd(A_i, B_j)</math> 加到另一个数组 <math>V</math> 中。对于 <math>U</math> 与 <math>V</math> 中不互质的元素连边，实际上我们需要求的就是这样二分图的一个最大匹配。但是这样边数比较多 (<math>O(n^4)</math>)，可以在中间建一些代表出现过的每个素数的临时点优化连边。再用网络流算法求最大流即可。</p>
时空复杂度	时间： $O(Maxflow(n^2, n^2 * \sqrt{n}))$ 空间： $O(n^2 + \sqrt{n})$

试题编号	Codechef Jul 14
名称	Sereja and Equality
题目大意	<p>给两个长度为 <math>n</math> 的数组 <math>A, B</math>，对于所有 <math>i(1 \leq i \leq n)</math>，满足 <math>C(A, A_i) = C(B, B_i)</math>，称这两个数组相似。其中 <math>C(X, x)</math> 等于满足 <math>X[j] &lt; x(1 \leq j \leq n)</math> 的 <math>j</math> 的数目。对于两个排列 <math>P1, P2</math>，定义一个函数 <math>F(P1, P2)</math> 等于满足 <math>P1[l \dots r]</math> 相似于 <math>P2[l \dots r](1 \leq l \leq r \leq n)</math> 并且 <math>P1[l \dots r]</math> 包含不超过 <math>E</math> 个逆序对的数对 <math>(l, r)</math> 的数目。求对 <math>P1, P2</math> 取遍所有 <math>n</math> 个元素的排列 <math>F(P1, P2)</math> 的总和是多少。<math>T \leq 10000, n \leq 500, E \leq 10^6</math>。</p>
算法讨论	<p>设 <math>cnt_{x,E}</math>，表示长度为 <math>x</math> 的排列中，逆序对为 <math>E</math> 的排列个数。可以使用前缀和优化的技巧 DP。于是答案 =</p> $\sum_{P1 \in Perm(n)} \sum_{P2 \in Perm(n)} \sum_{1 \leq l \leq r \leq n} [f(P1[l \dots r]) = f(P2[l \dots r])][inv(f(P1[l \dots r])) \leq E]$ $= \sum_{1 \leq l \leq r \leq n} \sum_{p \in Perm(r-l+1)} [inv(p) \leq E] \left( \sum_{P1 \in Perm(n)} [f(P1[l \dots r]) = p] \right)^2$ $= \sum_{x=1}^n (n-x+1) \sum_{p \in Perm(x)} [inv(p) \leq E] \left( \frac{n!}{x!} \right)^2$ $= \sum_{x=1}^n (n-x+1) \left( \sum_{i=0}^E cnt_{x,i} \right) \left( \frac{n!}{x!} \right)^2$ <p>其中 <math>f(p)_i = C(p, p_i) + 1</math>。 <math>inv(p)</math> 为 <math>p</math> 的逆序对数。</p>
时空复杂度	时间： $O(n^3)$ 空间： $O(n^3)$

试题编号	Codechef Jan 12
名称	Misinterpretation 2
题目大意	对于一个串 $S = s_1 s_2 s_3 s_4 \dots s_n$ , 规定 $f(s_1 s_2 s_3 \dots s_n) = s_2 s_4 s_6 \dots s_{[n/2]*2} s_1 s_3 s_5 \dots s_{[(n-1)/2]*2+1}$ 。求有多少长度在 $[L, R]$ 的小写字母构成的字符串 $S$ , 满足 $f(S)=S$ 。 $1 \leq T \leq 5, 1 \leq L \leq R \leq 10^4, R - L \leq 50000$ 。
算法讨论	可以认为 $f$ 函数是一个置换。根据 Polya 计数法若长度为 $n$ 的 $f$ 的循环数为 $k(n)$ , 那么长度为 $n$ 的满足条件的字符串的数量就应该是 $26^{k(n)}$ 。 $n$ 为奇数时, 最后一位不变, 所以有 $k(n) = k(n-1) + 1$ 。考虑 $n$ 为偶数时, 设 $\text{ord}(d)$ 为 2 模 $d$ 的阶。对 $1 \leq i \leq X$ , 可以证明 $i$ 最少乘 $\text{ord}((X+1)/\gcd(i, X+1))$ 个 2 余 $i$ 同余 ( $\text{mod } X+1$ )。并且这些数一共有 $\phi(p)$ 个, 所以构成 $\phi(p)/\text{ord}(p)$ 个循环。所以 $k(X) = \sum_{p (X+1), p \neq 1} \phi(p)/\text{ord}(p)$ , 可以发现对于 $\gcd(x,y)=1, \text{ord}(xy)=\text{lcm}(\text{ord}(x), \text{ord}(y))$ , 通过筛法求出 $L$ 到 $R$ 每个数的质因数, 现在还要求出 2 模所有指数幂的阶, 求 $\text{ord}(x)$ 可以用分解 $\phi(x)(=x-1)$ 的经典做法。于是就可以计算出 $k(X)$ 了。
时空复杂度	时间: $O(R^{3/4}/\log^2(R) + \sqrt{R}\log^2(R) + T(\sqrt{R}/\log(R) + \log^2(R)))$ 空间: $O(\sqrt{R})$

试题编号	Codechef Jan 12
名称	Card Shuffle
题目大意	维护一个 $n$ 张牌的牌堆, 从顶到底编号 1 到 $n$ , 重复 $M$ 次如下操作: 从牌堆顶拿走 $A$ 张牌, 再从牌堆顶拿走 $B$ 张牌, 将第一步拿走的 $A$ 张牌放回到剩下的牌堆上面。从牌堆顶拿走 $C$ 张牌。将第二步你拿起的 $B$ 张牌一张一张放到牌堆顶, 最后, 将剩下的 $C$ 张牌放回到牌堆顶。 $1 \leq N, M \leq 10^5, 0 \leq A_i, C_i < N, 0 < B_i, A_i + B_i, C_i + B_i \leq N$ 。
算法讨论	用一棵平衡二叉树模拟上述操作即可, 打一下翻转标记。
时空复杂度	时间: $O(M * \log(N))$ 空间: $O(N)$

试题编号	Codechef Jun 14
名称	Sereja and Arcs
题目大意	有 $N$ 个点，它们的坐标分别为 $(1,0),(2,0),\dots,(N,0)$ 。每个点有一个颜色，坐标为 $(i,0)$ 的点的颜色为 $A[i]$ 。如果有 $A[i] = A[j]$ 且 $i < j$ ，那么 Sereja 会画一条圆弧连接 $(i, 0)$ 和 $(j, 0)$ ，这条圆弧的颜色为 $A[i]$ ，圆心在 $X$ 轴上，且所有的圆弧都在第一象限内。问有多少对不同颜色的圆弧相交了。 $1 \leq N \leq 10^5, 1 \leq A[i] \leq 10^5$ 。
算法讨论	设 $cnt_i$ 为第 $i$ 种颜色的总点数，记 $pre_{i,j}$ 为前 $i$ 个点中具有颜色 $j$ 的点数。两种颜色的圆弧一共有 3 种情况：AABB，ABBA，ABAB。设他们的数量分别为 $s_1, s_2, s_3$ ，总的双色圆弧对有 $s$ 个，则有 $s_3 = s - s_1 - s_2$ ，而 $s_3$ 为所求。 $s = \sum_{i < j} \binom{cnt_i}{2} \binom{cnt_j}{2}$ 。对于 AABB 的 $s_1$ ，设 $K = \sum_{j \neq a_i} \binom{pre_{i,j}}{2}$ 表示 $i$ 左侧与其异色的圆弧数， $i$ 对 $s_1$ 的贡献为 $K * (cnt_{a_i} - pre_{i,a_i})$ 。对于 ABBA 的 $s_2$ ，我们对 $cnt_i$ 以 $k$ 为界进行划分并分别处理。当 $cnt_A \leq k$ ， $A$ 最多只有 $\frac{n}{k}$ 种，可以利用 $pre_{i,A}$ 的前缀和处理。当 $cnt_A < k$ 且 $cnt_B \geq k$ 时，每个颜色对的贡献是 $\sum_{i < j} \binom{x_j - x_i}{2}$ ， $x_i$ 为 $pre_{pos_i,B}$ ，化简后可以 $O(\frac{n^2}{k})$ 计算。当 $cnt_A < k$ 且 $cnt_B < k$ 时，对于一条具有颜色 $B$ 的圆弧 $(x,y)$ ，从小到大枚举左端点 $x$ ，然后再枚举右端点 $y$ ，可以通过处理后缀和来快速计算。复杂度为 $O(nk \log n)$ 。
时空复杂度	时间： $O(n * \sqrt{n * \log(n)})$ 空间： $O(n)$

试题编号	Codechef Sep 12
名称	Simultaneous Nim
题目大意	给定 $n$ 个 Xor 和为 0 的数 $A_i$ ，求划分成最多的份数，每份 Xor 和都为 0。 $10 \leq N \leq 1000, 5 \leq M \leq 60, 1 \leq A_i \leq 2^M - 1$ ，数据合法。
算法讨论	设 $bit$ 为 $A_i$ 的二进制位数。考虑贪心算法，每次删去一个包含最小的 $A_i$ 的 Xor 和为 0 的集合，这样最后得到的是一个较优的解。最小集合的求法可以考虑每个数是选或者不选，然后高斯消元，注意到最多只会 $bit$ 个方程，但是会有 $n$ 个变量。可以考虑随机出一些变量标记为可选，其他标记为不可选，判断是否有解，如果无解，尝试选更多的变量。
时空复杂度	时间： $O(Tnbit)$ 空间： $O(n)$

试题编号	Codechef Sep 12
名称	Annual Parade
题目大意	一个 $n$ 个点， $m$ 条边的有向图，边带权，现在要找出若干条路径，首先花费所有路径上的边的权值之和（若某条边出现在 $k$ 条路径中则计算 $k$ 次），随后如果有一条路径的起点和终点不同额外花费 $C$ 的费用，如果有一个点没有出现在任意一条路径中额外花费 $C$ 的费用。现有 $Q$ 组询问，每次给定一个 $C$ 的取值，问这种情况下的最小花费。 $n \leq 250, m \leq 30000$ 。
算法讨论	首先 floyd 预处理出两两之间的最短路，然后按照 DAG 求最小路径覆盖的模型建图。这样每次增光一个流量，出了消耗增光的费用，还会少用 $C$ 的费用。这是一个分段函数。预处理出每次增广的费用，然后根据询问的 $C$ 二分回答。
时空复杂度	时间： $O(n^3 + \text{Maxflow}(n, n^2) + Q * \log(n))$ 空间： $O(n^2)$

试题编号	Codechef Sep 12
名称	Knight Moving
题目大意	给一张无限大的方格棋盘。一个骑士从 (0,0) 开始，每次可以移动一个 $(Ax, Ay)$ 向量，或是一个 $(Bx, By)$ 向量。有 $K$ 个障碍格不能进入。计算到达 $(X, Y)$ 有多少种方案 (模 1000000007)。 $1 \leq T \leq 5, 0 \leq K \leq 15,  X ,  Y  \leq 500$ 。
算法讨论	如果 $A$ 和 $B$ 两个向量线性相关。可以转化为一维问题，可以暴力 DP 解决。否则每个点可以表示成 $a*A + b*B$ ，用组合数学 + 容斥原理可以解决。
时空复杂度	时间： $O(K^2 * 2^K + 500^2)$ 空间： $O(500^2 + K)$

试题编号	Codechef Sep 11
名称	Short
题目大意	给你两个数 $n, l$ ，求数对 $(a, b)$ 的数量，满足 $n < a < l, n < b < l$ ，并且 $ab - n$ 可以被 $(a - n)(b - n)$ 整除。 $0 \leq n \leq 10^5, k \leq 10^18$ 。
算法讨论	可以设 $c = a - n, d = b - n$ 。原题可改为，求 $c, d$ 使得 $cd   (c + n)(d + n) - n$ ，且 $1 \leq c, d < l - n$ 。可以解出 $d = \frac{cn + n^2 - n}{kc - c - n}$ 。由于 $n$ 的范围很小，所以可以枚举 $n$ ，然后若枚举了 $c$ ，可以算出 $cn + n^2 - n$ ，枚举其约数判断即可。（若假设 $c < d$ ，可以证明 $c \leq n * 3$ 。并且当 $c$ 较大时，比如大于 4000，可以改为枚举 $k$ 。）
时空复杂度	时间： $O(Tn4000)$ 空间： $O(n \log n)$



试题编号	Codechef Sep 11
名称	Counting Hexagons
题目大意	有一堆木棍，长度在 $[1,n]$ ，每种长度的木棍都有很多根。计算这些木棍可以拼多少种六边形。满足最长的木棍长度大于等于 $L$ ，其他木棍长度都小于 $X$ ，同样长度的木棍不能超过 $K$ 根。 $2 \leq N \leq 10^9, 2 \leq L \leq N, N - L \leq 100, 1 \leq X < L, 1 \leq K \leq 5$ 。
算法讨论	可以枚举最长边的长度 $M$ ，剩下 5 条边需满足和大于 $M$ ，最大的数小于 $M$ ，且相同的个数小于 $K$ 。可以用数位 DP 解决， $dp[i][s][c][p1][p2][p3]$ 表示当前枚举到第 $i$ 位，从大到小两两之间是否相等用二进制表示为 $s$ ，当前和是否已经小于 $S$ ，数列最大值是否小于 $M$ ，最小值是否每一位都是 0。枚举转移 DP 即可。
时空复杂度	时间： $O((N - L) * \log(N)2^4)$ 空间： $O((N - L) * 2^4)$

试题编号	Codechef Sep 14
名称	Rectangle Query
题目大意	维护二维平面上的矩形。1.I $x1\ y1\ x2\ y2$ 插入一个左下角在 $(x1,y1)$ ，右上角在 $(x2,y2)$ 的矩形。2.D $index$ 删除第 $index$ 次插入的矩形。3.Q $x1\ y1\ x2\ y2$ ，询问有多少之前加入的矩形，与左下角在 $(x1,y1)$ ，右上角在 $(x2,y2)$ 的矩形有至少一个公共点。 $1 \leq Q \leq 10^5, 1 \leq x1 \leq x2 \leq 10^9, 1 \leq y1 \leq y2 \leq 10^9$ 。
算法讨论	补集转化，求出 $m$ 没有公共点的矩形个数，再考虑容斥，变成求 $x$ 坐标不相交的个数 + $y$ 坐标不相交的个数 - 两个坐标都不相交的个数。通过整理式子可以发现这是一个三维偏序的问题。可以用 CDQ 分治加数据结构解决。
时空复杂度	时间： $O(Q * \log^2(Q))$ 空间： $O(Q)$

试题编号	Codechef Sep 14
名称	Fibonacci Numbers on Tree
题目大意	维护一棵 $n$ 个节点带点权的有根树。初始每个节点权值为 0，有 $m$ 个操作。1. $A\ x\ y$ : 将 $x$ 到 $y$ 路径上第 $k$ 个点的权值加上 $F[k]$ 。2. $QS\ x\ y$ : 视 $x$ 为根，询问以 $y$ 节点为根的子树中所有节点的权值和。3. $QC\ x\ y$ : 询问 $x$ 到 $y$ 路径上所有节点的权值和。4. $R\ x$ : 将所有节点的权值还原到第 $x$ 个操作后的状态。（ $F[k]$ 为第 $k$ 个斐波那契数。）强制在线。输出对 1000000009 取模。 $1 \leq n, m \leq 10^5, 1 \leq x, y \leq n$ 。
算法讨论	这个题目可以参考 Observing the Tree，区别在于链上加等差数列变成了链上加斐波那契数列，还有一个换根查子树的操作。维护斐波那契数列可以与维护等差数列类似，如果只维护前两项，与等差数列类似也是可以快速合并的。知道斐波那契前两项后求第 $n$ 项可以用斐波那契的通项公式，涉及到 $\sqrt{5}$ ，在模 1000000009 的意义下可以用它的二次剩余代替。连续斐波那契数的和可以表示成另一个斐波那契数。换根查子树和的操作可以讨论新的根与要查询节点的关系，分别对应三种情况：1. 查询的还是以原来根的子树，2. 查询的是整棵树（ $x$ 与 $y$ 相等时），3. 查询的是以原来根的子树的补集。分别判断就可以。
时空复杂度	时间： $O(m * \log^2(n))$ 空间： $O(m * \log^2(n))$

试题编号	Codechef Feb 13
名称	Room Corner
题目大意	给一个用 +,-, , 空格描述的 $n*m$ 的房间，保证每行房间连续，每个拐角处有一个小朋友，每个小朋友可以与相邻的小朋友交换。每走过一个格子，消耗 1 的时间。T 个询问，询问两个小朋友相遇的最短时间。 $T \leq 10^4, n, m \leq 2500$ 。
算法讨论	在对图进行处理后，可以转化为一个环上的问题，环上有 $n$ 个小朋友，每两个相邻的小朋友直接都有一段距离。问环上两个点通过交换相遇的最短时间。最后肯定是两个小朋友交换到了一对相邻的位置，然后相遇。这部分可以通过二分时间解决。然后再对环上的两种走法取 min 即可。
时空复杂度	时间： $O(nm + T * \log(n))$ 空间： $O(nm)$

试题编号	Codechef Mar 14
名称	Chef and Graph Queries
题目大意	有一个无向图 G。顶点从 1 到 N 标号，边从 1 到 M 标号。有 Q 对询问 $L_i, R_i$ ( $1 \leq L_i \leq R_i \leq M$ )。对于每对询问，大厨想知道当仅保留编号 X 满足 $L_i \leq X \leq R_i$ 所在的边时候，图 G 中有多少连通块。 $1 \leq T \leq 1000, 1 \leq N, M, Q \leq 200000, 1 \leq U_i, V_i \leq N, 1 \leq L_i \leq R_i \leq M$ 。
算法讨论	对于每条边，我们认为这条边的权值是这条边的编号。对于询问，可以按 R 分类，从小到大回答，对于编号 1~R 的边维护一棵最大生成树，如果这棵树上 $L \sim R$ 一共有 k 条边，那么原图就会有 $(n-k)$ 个连通块。可以用 LCT 维护最大生成树，树状数组维护当前最大生成树的编号。
时空复杂度	时间： $O(n * \log(n))$ 空间： $O(n + m)$

试题编号	Codechef Mar 14
名称	The Street
题目大意	<p>一条大街上有 <math>N</math> 个商店，编号 1 到 <math>N</math>，每个商品的花费为商品花费 + 税费，初始所有商店没有商品，且税费为 0。需要维护以下操作：1. <math>u\ v\ a\ b</math>: 新增商品操作，对于编号 <math>u</math> 到 <math>v</math> 的商店，新增花费为首项为 <math>b</math>，公差为 <math>a</math> 的等差数列的商品。2. <math>u\ v\ a\ b</math>: 税费调整操作，对于编号 <math>u</math> 到 <math>v</math> 的商店，税费增加首项为 <math>b</math>，公差为 <math>a</math> 的等差数列。3. <math>i</math>: 询问操作，询问顾客在第 <math>i</math> 个商店购买一个商品的最高花费。</p> <p><math>1 \leq N \leq 10^9, 1 \leq M \leq 3 * 10^5, 1 \leq u \leq v \leq N, 1 \leq i \leq N</math>, 对于 1 操作 <math> a ,  b  \leq 10^9</math>，对于 2 操作 <math> a ,  b  \leq 10^4</math>。</p>
算法讨论	<p>首先可以把商店编号离散化到 <math>O(m)</math>。对于税费，可以用一个线段树维护不同等差数列的和，这个是可以根据首项和公差合并的。对于商品价格可以用线段树维护等差数列的最值，用优势区间算法，每个线段树的节点保存在这个区间最可能称为最值的等差数列，查询时，对 <math>\log(m)</math> 个等差数列上这个位置的值取 <math>\max</math> 即可。</p>
时空复杂度	时间: $O(m * \log(m))$ 空间: $O(m)$

试题编号	Codechef May 15
名称	Counting on a directed graph
题目大意	<p>给定一个 <math>N</math> 个点 <math>M</math> 条边的有向图。请你统计无序对 <math>(X,Y)</math> 的个数，其中 <math>(X,Y)</math> 满足存在一条从点 1 到点 <math>X</math> 的路径，和一条从点 1 到点 <math>Y</math> 的路径，且两条路径除了点 1 以外没有公共点。<math>1 \leq n \leq 10^5, 0 \leq m \leq 5 * 10^5</math>。</p>
算法讨论	<p>根据有向图必经点的算法建出 dominator-tree，原题的 <math>X, Y</math> 只要不来自 1 的同一个儿子及合法，扫一遍统计即可。</p>
时空复杂度	时间: $O((n + m) * \alpha(n))$ 空间: $O(n + m)$

试题编号	Codechef May 15
名称	Chef and Balanced Strings
题目大意	<p>一个字符串 <math>S</math> 如果能分成两个多重集和 <math>M1, M2</math> 且 <math>M1=M2</math> 称为 <math>S</math> 是平衡的。给定字符串 <math>T</math>，每次询问 <math>L, R, type</math> 求：</p> $\sum_{L \leq s < e \leq R, T_{s,e} \text{ 是平衡的}}  T_{s,e} ^{type}$ <p>。 <math> S </math> 表示 <math>S</math> 的长度。强制在线。 <math>1 \leq T \leq 10^5, 1 \leq N \leq 10^5, 1 \leq Q \leq 10^5, 0 \leq X, Y \leq N, 0 \leq type \leq 2</math>。</p>
算法讨论	<p>首先，<math>S</math> 是平衡的可以理解为 <math>S</math> 中每个字母出现了偶数次。对每个字母分配一个不同的 2 的幂，表示出现次数的奇偶，设为 <math>x_i</math>。若 <math>x_{L-1} = x_R</math> 可以认为，<math>L</math> 到 <math>R</math> 的子串是平衡的。对 <math>T</math> 的下标分成 <math>\sqrt{n}</math> 块，预处理出 <math>f[i][j]</math>，前 <math>i</math> 个块里 <math>x_k = j</math> 的个数，以及 <math>g[i][j]</math>，前 <math>i</math> 个块里 <math>x_k = j</math> 的下标和，以及 <math>h[i][j]</math>，前 <math>i</math> 个块里 <math>x_k = j</math> 的下标平方和，以及 <math>ans[i][j]</math>，表示第 <math>i</math> 块到第 <math>j</math> 块的答案。对于每次询问 <math>L, R</math>，根据完整包含的块可以分成三段。没有完整包含的部分长度一定小于 <math>\sqrt{n}</math>，可以开一个数组扫一遍并结合 <math>f, g, h</math> 可以累加出答案。</p>
时空复杂度	时间： $O(n * \sqrt{n})$ 空间： $O(n * \sqrt{n})$

试题编号	Codechef Oct 13
名称	Fibonacci Number
题目大意	<p>斐波那契数定义为, <math>F_0 = 0, F_1 = 1, F_i = F_{i-1} + F_{i-2} (i &gt; 1)</math>。给定一个素数 <math>P</math>，和非负整数 <math>C</math>，找到最小的非负整数 <math>n</math>，满足 <math>F_n = C \pmod{P}</math>。 <math>1 \leq T \leq 100, 11 \leq P \leq 2 * 10^9, P</math> 是素数, <math>0 \leq C \leq P - 1, (P \pmod{10})</math> 是完全平方数 保证，若答案存在，不会超过 <math>2 * 10^9</math>。</p>
算法讨论	<p>可以根据特征根法算出 <math>F</math> 的通项公式，发现题目保证了 5 有二次剩余，化简完方程发现 <math>\phi^n - \phi^{-n} = a</math>，通过讨论 <math>n</math> 的奇偶可以解出 <math>\phi^n</math>。然后需要用二次剩余判断是否有合法解。若存在可以通过 BSGS 算出 <math>x^n</math> 的最小的 <math>n</math>。否则无解。求解二次剩余可以参考：<a href="http://en.wikipedia.org/wiki/Cipolla%27s_algorithm">http://en.wikipedia.org/wiki/Cipolla%27s_algorithm</a></p>
时空复杂度	时间： $O(\sqrt{P})$ 空间： $O(\sqrt{P})$

试题编号	Codechef Jun 12
名称	Expected Maximum Matching
题目大意	按以下方式随机生成一个二分图：左边第 $i$ 个点和右边第 $j$ 个点之间有边的概率为 $f[i][j]$ 。求这样生成的二分图的最大匹配的期望值。左边有 $n$ 个点，右边有 $m$ 个点。 $1 \leq n \leq 5, 1 \leq m \leq 100, 0 \leq p[i][j] \leq 1, p[i][j]$ 均为五位小数。
算法讨论	根据 Hall 定理，左边 $k$ 个点能匹配的充要条件是这 $k$ 个点的任意一个子集 $s$ ，右边都有至少 $ s $ 个点临界。可以对左边所有的子集状压成一个状态，右边从 1 到 $m$ 去扫，枚举每条边连不连。DP 出所有的情况对应的概率。
时空复杂度	时间： $O(m * 2^{2^n})$ 空间： $O(2^{2^n})$

试题编号	Codechef Jun 12
名称	Cool Numbers
题目大意	一个数有 $k$ 位，每个数位上的数字分别为 $X_1, X_2, \dots, X_k$ ，如果一个数 $n$ 存在一到三个数位上的数的和为 $S$ ，且 $(X_1 + X_2 + \dots + X_k - S)^S$ 是 $n$ 的倍数，那么 $n$ 是 cool number。定义 $LC(N)$ 和 $RC(N)$ 分别是小于等于 $N$ 的最大的 cool number 和大于 $N$ 的最小的 cool number，多次询问给定 $N$ ，求 $LC(N)$ 和 $RC(N)$ 。 $1 \leq T \leq 100000, 1 \leq N \leq 10^{1000}, \sum N \leq 10^{4000000}$ 。
算法讨论	如果一个数有 1 ~ 3 个数位上非 0，让这三个数位的和成为 $S$ 即合法。否则有 $(9s)^{27} \geq 10^{s-1}$ ，这样的 $s$ 不会超过 80，可以全部找出，一共不到 40000 个。计算答案可以分别考虑两位数，第一类可以构造出，第二类可以在所有数里二分。然后比较两次求出答案。
时空复杂度	时间： $O(T * \log(N))$ 空间： $O(\log(N))$

试题编号	Codechef Jun 12
名称	Closest Points
题目大意	bytetown 在三维空间中，并包含 $n$ 个响应中心和 $q$ 个银行。你需要为每个银行找到离他欧几里得距离最近的响应中心，如果有多个响应中心，你可以任意输出其中的一个。注意一个响应中心可以对应多个银行。你不需要每个输出都是正确的。 $n = q = 50000$ 。
算法讨论	求最近点可以用 K-d Tree，因为题目数据范围太大，没有办法全部都算出来，加一个变量统计运算次数，然后卡时，可以得到较优的解。
时空复杂度	时间： $O(q * n^{\frac{2}{3}})$ 空间： $O(n)$

试题编号	Codechef Aug 12
名称	Two Magicians
题目大意	有一张 $n$ 个点， $m$ 条边的简单无向图，A 和 B 博弈，开始时 A 在点 1，B 在点 2，A 先手，每次一个人有三个操作： 1. 可以沿着现有的无向边任意移动，如果这一步结束时两个人在一个点，当前人胜。 2. 加入一条连接两个节点现在还没存在的无向边，如果无法加入则另一个人获胜。 3. 最开始每个人有 $P$ 次传送机会，如果当前人还有传送机会，他可以选择消耗一次并传送到任意一个节点。问谁必胜。 $T \leq 100, n \leq 7777, m, P \leq 10^5$ 。
算法讨论	可以 DP 做，当前需要记录的状态有两人所在连通块大小的奇偶性，两个人剩下的传送次数，奇数块的个数，偶数块的个数以及添加后不影响连通性的边的条数。枚举最后一次操作后可能的转移即可。但复杂度较高。通过打表观察，发现 DP 值是在一些特定情况是循环的。可以简化状态。然后用带权并查集维护连通块奇偶即可。
时空复杂度	时间： $O(n\alpha(n))$ 空间： $O(n + m)$

试题编号	Codechef Aug 12
名称	A Game of Thrones
题目大意	有 $N$ 个数字写在一张纸上（可能有相同数字），双方轮流 A 先手，在第一轮中，A 先选择一个纸上的数字，把它称作当前数字。之后从第二轮开始执行当前回合的人都按如下操作：我们称当前数字为 $u$ 。将 $u$ 从纸上擦去（如果有多个擦去任意一个）。然后选择另一个纸上的数字 $v$ 作为当前数字， $v$ 要满足 $u$ 刚好相差一个质因子。无法完成操作的人输。无法完成操作的人输。如果先手必胜还需输出在第一轮里能保证必胜的最小的数。 $1 \leq N \leq 500, 1 \leq u \leq 10^4, 1 \leq c \leq 10^9, c$ 是某个数字的个数。
算法讨论	发现相差一个质因子的数字构成了一个二分图。如果这个二分图存在完备匹配，后手必胜，否则先手必胜。可以建出图，用网络流判断。先手必胜时，一定是除了某个数存在完备匹配，从小到大枚举数字，然后用退流判断。
时空复杂度	时间： $O(N^2 * \log(u) + \text{Maxflow}(n, n * \log(n)))$ 空间： $O(n * \log(n))$

试题编号	Codechef Jan 14
名称	Counting D-sets
题目大意	求 $N$ 维空间中的整点，满足其直径恰好等于 $D$ 。点集的直径是点集中最远的一对点的切比雪夫距离。两个点集称为相等的。当且仅当点数相等，并且一个点集中的点都移动一个向量后与第二个点集中的点相同。答案对 $10^9 + 7$ 取模。 $1 \leq T \leq 10, 1 \leq N \leq 1000, 1 \leq D \leq 10^9$ 。
算法讨论	<p>首先可以把所有 <math>N</math> 维点集通过平移使至少一维等于 0。可以令 <math>cnt_i</math> 表示直径小于 <math>i</math> 的点集数量。直径等于 <math>D</math> 的点集数量就是 <math>cnt_D - cnt_{D-1}</math>。设 <math>f_i</math> 为直径为 <math>D</math> 的 <math>1 \sim i</math> 维都 <math>&gt;0</math> 的点集数量。有</p> $f_i = 2^{D^i * (D+1)^{n-i}} - \sum_{i < j \leq N} C_{j-i}^{n-i} * f_j$ <p>则 <math>cnt_D = f_0</math>。</p>
时空复杂度	时间： $O(N^2)$ 空间： $O(N^2)$



试题编号	Codechef Jan 14
名称	Counting The Important Pairs
题目大意	给你一个 $n$ 个点 $m$ 条边的无向连通的简单图。求删两条边后仍连通的方案数。 $1 \leq N \leq 100000, 1 \leq M \leq 300000$ 。
算法讨论	随便找出一个生成树，给每条非树边一个随机非负权值，每条树边的权值是跨越这条树边权值的 xor 和。删掉两条边不连通有两种情况：一种是两条边 xor 值为 0，另一种是其中一条边的权值为 0。分别统计即可。
时空复杂度	时间： $O(N + M)$ 空间： $O(N + M)$

试题编号	Codechef Mar 13
名称	Little Elephant and Colored Coins
题目大意	有 $N$ 种硬币，标号 1 到 $N$ ，第 $i$ 种硬币价值 $V_i$ 美元，颜色是 $C_i$ ，对于每种硬币，都有无限个。多次询问能不能恰好组成 $S$ 美元，如果能最多能有多少种颜色。 $1 \leq N \leq 30, 1 \leq V_i \leq 2 * 10^5, 1 \leq C_i \leq 10^9, 1 \leq Q \leq 2 * 10^5, 1 \leq S \leq 10^{18}$ 。
算法讨论	首先不考虑颜色。任取一个 $V_i$ 记为 $M$ ，记 $dp_i$ 为最小的 $x$ ，使得 $x \bmod M = i$ ，且 $x$ 能够由其他的 $V_i$ 组成。有转移 $dp_x + V_i \rightarrow dp_{(x+V_i) \bmod M}$ 。用当前 $dp$ 数组就可以判断是否能恰好组成 $S$ 美元，但复杂度较高。考虑状态 $f_{i,j}$ 为只考虑前 $j$ 种硬币得到的 $dp_i$ 。有转移 $f_{i,j} \leftarrow V_j + f_{i-V_j \bmod M, j-1}$ 。发现同一个 $j$ 的转移是一些环，这个环可以从最小的 $f_{i,j}$ 断开，然后用单调队列做。复杂度降为 $O(NM)$ 。如果考虑颜色可以在状态里多加一维并且同样颜色一块 DP，特判 $M$ 的颜色即可。
时空复杂度	时间： $O(N^2 * M)$ 空间： $O(Q * N)$

试题编号	Codechef Mar 12
名称	Evil Book
题目大意	有 $N$ 个厨师，对于第 $i$ 个厨师可以付出 $C_i$ 点努力打败他并得到 $M_i$ 点魔力，同时 $M_i$ 变为 0。在拥有魔力不少于 $X$ 时可以使用帮助，即通过消耗 $X$ 点魔力使某个厨师的 $C_i$ 和 $M_i$ 都乘以 $\frac{1}{3}$ 。初始时拥有魔力为 0，求收集至少 666 点魔力所需要最少的总努力。 $1 \leq T \leq 5, 1 \leq N \leq 10, 10 \leq X < 666, 0 \leq C_i \leq 10^9, 0 \leq M_i \leq 10^9$ 。
算法讨论	有一个结论，“打败某个厨师获得的魔力必然大于对他使用帮助所消耗的魔力”，即对第 $i$ 位厨师使用了 $k$ 次魔力需满足 $\frac{M_i}{3^k} > kX$ 。还有一个结论对于第 $i$ 位厨师，令 $K$ 为最小的非负整数使得 $\frac{M_i}{3^K} < 666$ 。设当前在魔力的限制下最多能用 $can$ 次帮助，则对该位厨师使用的帮助次数 $k$ 满足 $k \geq \min(can, K - 1)$ 。然后可以用搜索加剪枝解决此题。
时空复杂度	时间： $O(?)$ 空间： $O(n)$

试题编号	Codechef Mar 12
名称	Ciel and Earthquake
题目大意	给一个 $R \times C$ 的网格图，每条边有 $p$ 的概率被删除。求 (1,1) 与 (R,C) 连通的概率。 $1 \leq T \leq 50, 1 \leq R \leq 8, 1 \leq C \leq 10^8, 0.1 \leq p \leq 1$ ， $p$ 小数点后最多四位。
算法讨论	若 $m$ 比较小，可以用插头 DP，状压边界每个点与 (1,1) 的连通性，预处理转移。当 $m$ 比较大的时候，DP 状态是一个近似等比数列。可以用公比计算。
时空复杂度	时间： $O(Tnm2^n)$ 空间： $O(nm2^n)$

试题编号	Codechef Jun 13
名称	Two k-Convex Polygons
题目大意	给定 $n$ 个棍子的长度和整数 $k$ ，求能否在其中选出 $2k$ 个棍子拼成两个凸多边形。使得两个凸多边形都恰好有 $k$ 跟棍子组成，且任意相邻的边都不共线。输出方案。 $2k \leq n \leq 1000, 3 \leq k \leq 10, 1 \leq L_i \leq 10^9$ 。
算法讨论	构成凸多边形只需要最长的边的长度大于其他边的和即可。把 $n$ 根木棍排序。若两个凸多边形用的在这个序列里不交叉，可以贪心选一个最靠前的和最靠后的，看是否相交。判断是否合法可以用前缀和。如果相交，那么构成两个凸多边形的木棍一定是连续的 $2k$ 个木棍。可以 $2^{2k}$ 枚举每根木棍属于哪个多边形然后判断。
时空复杂度	时间： $O(n * 2^k)$ 空间： $O(n)$

试题编号	Codechef Aug 13
名称	Prime Distance On Tree
题目大意	给定一棵树, 树上边长度均为 1。如果我们在树中等概率地选取两个不同的点, 这两个点之间的距离是一个质数的概率是多少呢? $2 \leq N \leq 50000$ 。
算法讨论	答案等于距离等于质数的选两点的方案数/ $C(N,2)$ 。可以用点分治加 FFT 求出每个距离出现的次数, 对于距离是素数的求和即可, 注意不要重复计数即可。
时空复杂度	时间: $O(N * \log(N))$ 空间: $O(N)$

试题编号	Codechef Aug 13
名称	Deleting numbers
题目大意	当前有 $n$ 个数形成的数组 $a_i$ , 每次可以选择 $v, t$ 两个数, 满足 $v + t \leq n + 1$ 且令 $k$ 为最大的满足 $v + kt \leq n$ 的数, 必须满足 $a_v = a_{v+t} = a_{v+2t} = \dots = a_{v+kt}$ 。然后这些数将会删除 (若 $v + t = n + 1$ 则只删除第 $v$ 个数), 之后形成新的由 $n - k$ 个数形成的数组 $a_i$ , 满足这个数组的元素是原来数组被删除之后剩下的元素, 且相对位置保持不变。设计一种方案删除所有的数, 使总步数尽量小。 $1 \leq n \leq 100000, 1 \leq a_i \leq 100000$ 。
算法讨论	单次枚举所有删除的方法复杂度是 $O(n * \log(n))$ 。可以多次这样贪心, 然后在时间不够时一个一个删除。能得到较优的解。
时空复杂度	时间: $O(n^2 * \log(n))$ 空间: $O(n)$

试题编号	Codechef Jul 15
名称	A game on a graph
题目大意	A 和 B 在无向图 G 上用硬币进行博弈。A 选择一个起始点，把硬币放在这个点上。接下来 A 和 B 轮流操作，B 先操作。轮到每个玩家时，他需要把硬币沿着一条边移至另一点。硬币不能到达过的点。无法操作者输。求 G 有多少个 A 能通过开始选择该点获胜的点。 $1 \leq T \leq 3, N-1 \leq M \leq \min(C(N, 2), 300000), 1 \leq N \leq 2000$ 。
算法讨论	胜利点的充要条件是：在任意一个图 G 的最大匹配中，点 u 为孤立点，也就是未被匹配。可以用带花树算法求出图 G 的一种最大匹配。可以从每个孤立点出发在做一次增广，把所有距离为偶数或者花中的节点都标记为必胜。然后统计点数即可。
时空复杂度	时间： $O(Tnm)$ 空间： $O(n+m)$

试题编号	Codechef Jul 15
名称	Easy Exam
题目大意	有一个 K 面的骰子，每个面上的数字分别是 1 到 K。同时给以两个参数 L 和 F ( $0 < L \leq K$ )。求投 N 次这个骰子。令 $a_i$ 为掷出数字 i 的次数。求 $a_1^F * a_2^F * \dots * a_L^F$ 的期望值。 $1 \leq T \leq 2, 0 < N, K \leq 10^9, 0 < L * F \leq 20000, 0 < F \leq 1000$ 。
算法讨论	设 $x_{i,j}$ 为第 j 次是否投到 i，若投到就是 1 否则就是 0。有 $a_i = \sum_{j=1}^n x_{i,j}$ ，答案就是 $\prod_{i=1}^L \left( \sum_{j=1}^n x_{i,j} \right)^F$ 。展开后每项可以独立计算。若一项中同时有 $x_{a,c}, x_{b,c}$ 那么这项贡献为 0。否则这项有 size 个不同的变量并且互相独立。期望值就为 $\frac{1}{K}^{size}$ 。然后需要计算的就是这些项的系数。令 $w_{i,j}$ 为式子 $(\sum x_k)^i$ 的展开中出现了 j 个不同变量的系数之和。有递推式 $w_{i,j} = w_{i-1,j-1} + w_{i-1,j} * j$ 。令 $dp_{n,m}$ 为原式展开中出现 j 个不同变量系数之和的方案数。有 $dp_{n,m} = dp_{n-1,j} * w_{F,m-j} * \binom{m}{j}$ 。可以利用指数生成函数，用 FFT+ 快速幂优化该递推式递推。
时空复杂度	时间： $O(F^2 + LF * \log(LF) * \log(L))$ 空间： $O(L * F)$

试题编号	Codechef Aug 13
名称	Music & Lyrics
题目大意	给定 $W$ 个模式串 $T_i$ , $N$ 个主串 $S_i$ , 找出其中每一个单词在所有的主串中出现的次数。 $1 \leq W \leq 500, 1 \leq  T_i  \leq 1000, 1 \leq N \leq 100, 1 \leq  S_i  \leq 50000$ 。
算法讨论	模式串建一个 AC 自动机。把所有主串在 AC 自动机跑一遍, 然后在 AC 自动机的 fail 树上 push 一遍。然后输出每个模式串对应节点的出现次数即可。
时空复杂度	时间: $O(W *  T_i  + N *  S_i )$ 空间: $O(W *  T_i  +  S_i )$

试题编号	Codechef Jun 13
名称	To challenge or not
题目大意	有 $M$ 个整数 $B_1, B_2, \dots, B_m$ , 你需要从中选出最多的数, 保证里面不存在长度为 3 的等差数列。 $N, B_i \leq 10^5$ 。数据随机。
算法讨论	贪心。将 $M$ 个整数排序, 从小到大判断, 如果加进来的这个数不作为最大数构成长度为 3 的等差数列, 就加进来。可以得到一个较优解。
时空复杂度	时间: $O(N^2)$ 空间: $O(N)$

试题编号	Codechef Jun 13
名称	Count Special Matrices
题目大意	<p><math>N</math> 是一个大于等于 3 的正整数。 <math>A</math> 是一个 <math>N * N</math> 的整数矩阵。满足以下条件, 称 <math>A</math> 是特殊的。</p> $A_{x,x} = 0 \text{ for } 1 \leq x \leq N$ $A_{x,y} = A_{y,x} > 0 \text{ for } 1 \leq x < y \leq N$ $A_{x,y} \leq \max(A_{x,z}, A_{z,y}) \text{ for } 1 \leq x < y \leq N$ $\forall k \in \{1, 2, \dots, N-2\} \exists x, y \in \{1, 2, \dots, N\} A_{x,y} = k$ <p>给出 <math>N</math> 求大小为 <math>N</math> 的特殊矩阵有多少。对 <math>10^9 + 7</math> 取模。 <math>1 \leq T \leq 10^5, 3 \leq N \leq 10^7</math>。</p>
算法讨论	<p>通过推倒可以得到答案等于</p> $\frac{N! * (N-1)!}{3 * 2^{N-1}} \left( \frac{3n}{2} - 2 - \sum_{i=1}^{n-1} \frac{1}{i} \right)$ <p>。预处理答案, 减少乘法次数计算。</p>
时空复杂度	时间: $O(N + T)$ 空间: $O(N)$

试题编号	Codechef Oct 11
名称	The Baking Business
题目大意	厨师在不同省份的不同城市甚至城市进一步划分成的地区出售产品。他还记录了每一次购买的顾客的性别和年龄。你需要处理这些信息，来接受查询。 $Q \leq 100000$ 。
算法讨论	根据题目模拟即可。
时空复杂度	时间： $O(N + Q)$ 空间： $O(N)$

试题编号	Codechef Jan 15
名称	Xor Queries
题目大意	给定一个初始时为空的整数序列 (元素由 1 开始标号) 以及一些询问：类型 0：在数组最后加入数字 $x$ 。类型 1：在区间 $L..R$ 中找到数字 $y$ ，最大化 $(x \text{ xor } y)$ , xor 表示按位异或。类型 2：删除数组最后 $k$ 个元素。类型 3：在区间 $L..R$ 中，统计小于等于 $x$ 的元素个数。类型 4：在区间 $L..R$ 中，找到第 $k$ 小的数 (第 $k$ 个顺序统计量)。 $1 \leq L \leq R \leq N, 1 \leq x \leq 5 * 10^5, 1 \leq M \leq 5 * 10^5$ 。
算法讨论	查询最大 xor 值以及小于等于 $x$ 的元素个数都可以用 Trie 来实现。区间操作可以用可持久化实现。
时空复杂度	时间： $O((N + M) * \log(N))$ 空间： $O(N * \log(N))$

试题编号	Codechef Oct 11
名称	Sine Partition Function
题目大意	<p>定义函数：</p> $f(M, N, X) = \sum_{k_1+k_2+\dots+k_M=N} \sin(k_1X)\sin(k_2X)\dots\sin(k_MX)$ <p><math>k_1, k_2, \dots, k_M</math> 都是非负整数。给出 M,N,X, 求 f(M,N,X)。保证答案不会超过 <math>10^{300}</math>。 <math>1 \leq T \leq 10, 1 \leq M \leq 30, 1 \leq N \leq 10^9, 0 \leq X \leq 6.28 &lt; 2 * \pi</math>。</p>
算法讨论	<p>考虑递推：</p> $dp_{i,j} = \sum_{k_1+k_2+\dots+k_i=j} \sin(k_1X)\sin(k_2X)\dots\sin(k_iX)$ <p>转移只需枚举最后 <math>k_i</math> 的值即可。<math>dp_{i,j} = \sum_{k_m=0}^j dp_{i-1,j-k_m} \sin(k_mX)</math> 直接递推复杂度 <math>O(MN^2)</math>。发现对于 <math>k_i \geq 2</math> 有 <math>\sin(k_iX) = \sin((k_i-1)X) = \sin((k_i-1)X)\cos(X) + \sin(X)\cos((k_i-1)X) = 2\sin((k_i-1)X)\cos(X) + \sin(X)\cos((k_i-1)X) - \sin((k_i-1)X)\cos(X) = \sin((k_i-1)X)2\cos(X) - \sin((k_i-2)X)</math>。 所以 <math>dp_{i,j} = dp_{i-1,j-1}\sin(X) + dp_{i,j-1}2\cos(X) - dp[i][j-2]</math>。 可以使用矩阵乘法优化转移。</p>
时空复杂度	时间： $O(M^3 * \log(N))$ 空间： $O(M^3)$

试题编号	Codechef Nov 11
名称	Luckdays
题目大意	<p>定义一个数列 <math>S_i</math>。 <math>S_1 = A, S_2 = B, S_i = (X * S_{i-1} + Y * S_{i-2} + Z) \bmod P</math>。每个询问 <math>L_i, R_i</math>，求满足 <math>L_i \leq k \leq R_i</math> 且 <math>S_k = C</math> 的 k 的个数。C 所有询问相同。 <math>1 \leq T \leq 2, 2 \leq P \leq 10007, 0 \leq A, B, X, Y, Z, C &lt; P, 1 \leq Q \leq 2 * 10^4, 1 \leq L_i \leq R_i \leq 10^{18}</math>。P 是一个素数。</p>
算法讨论	<p>可以把转移用一个 3*3 的矩阵记下来。然后可以对这个矩阵用 BSGS 算法。可以求出循环节。再求出 C 第一次出现的位置，就可以方便的回答询问了。</p>
时空复杂度	时间： $O(P * \sqrt{P} + Q * \log(P))$ 空间： $O(P)$



试题编号	Codechef Jan 15
名称	Ranka
题目大意	在一个 $9*9$ 的围棋棋盘上，构造一个 $n$ 步局面不重复的下法，玩家某轮中可以放弃。 $n \leq 10^4$ 。
算法讨论	先用黑子将棋盘下满，每下一颗黑子前都用白子将其余所有点填满（放下黑子就能把所有白子删掉），这样，大概能下 $9^4$ 次。反过来再用白子覆盖棋盘一次，就够 $1w$ 次了。
时空复杂度	时间： $O(n)$ 空间： $O(1)$

试题编号	Codechef Oct 14
名称	Children Trips
题目大意	给一棵 $n$ 个点的树，树上每条边的长度为 1 或 2。有 $M$ 个询问，给 $S, F, P$ 。求 $S$ 到 $F$ 需要多长时间。每个单位时间最多走 $P$ ，并且在一个单位时间结束的时候必须在某个节点上。 $1 \leq n, M \leq 10^5, 1 \leq X, Y, S, F \leq N, 2 \leq P \leq 2 * N$ 。
算法讨论	考虑对 $P$ 分块， $P$ 小于 $\sqrt{N}$ 的一块做，最多有 $\sqrt{N}$ 种，可以通过倍增的方法计算每个点往上走 $2^j$ 单位时间到哪个点。对于 $P$ 大于 $\sqrt{N}$ 的答案是 $O(\sqrt{N})$ 的。可以统一倍增出每个点走 $2^j$ 距离到哪里。然后根据 LCA 分成两段，分别模拟，中间特判即可。
时空复杂度	时间： $O(N * \sqrt{N} * \log(N))$ 空间： $O(N * \log(N))$

试题编号	Codechef Oct 12
名称	Max Circumference
题目大意	给出一个三角形 ABC，以及 N 个操作。第 i 个操作有两个参数 $x_i, y_i$ ，使用这个操作可以使得点 A 的 x 坐标增加 $x_i$ ，并且 y 坐标增加 $y_i$ 。你可以使用最多 K 个操作，这些操作的影响叠加，同一个操作不能重复使用，ABC 三个点允许共线或重合。最大化三角形 ABC 的周长，答案的绝对误差必须小于 $10^{-12}$ 。 $K \leq N \leq 500,  A_x ,  A_y ,  B_x ,  B_y ,  C_x ,  C_y  \leq 10^9,  x_i ,  y_i  \leq 10^6$ 。
算法讨论	可以证明最大化三角形的周长，等价于最大化 $f(X,Y)=uX+vY$ 。如果有了一组 (u,v)，可以取最多前 K 大的非负贡献的操作。对于任意两个操作 $A(x_1, y_1)$ 和 $B(x_2, y_2)$ 。存在某个区间 A 的贡献大于 B 的贡献，和某个区间 B 的贡献大于 A 的贡献。把所有操作两两求出 $N^2$ 段，就能代表所有情况。然后根据极角序按顺序扫，每次会交换两个操作。题目精度要求较高，极角序可以分象限叉积排，开方可以根据整数部分再精确计算小数部分。
时空复杂度	时间： $O(N^2 * \log(N))$ 空间： $O(N^2)$

试题编号	Codechef Nov 12
名称	Arithmetic Progressions
题目大意	给定 N 个整数 $A_1, A_2, \dots, A_N$ ，求有多少个三元组 (i,j,k) 满足 $A_i + A_k = 2 * A_j$ 。 $3 \leq N \leq 10^5, 1 \leq A_i \leq 3 * 10^4$ 。
算法讨论	对数列分成 T 块，枚举每一块。有三种情况：三个数都在当前块，在当前块内枚举前两个数字，然后查询块内第三个数字数量；两个数在当前块，与上面类似，查询换为整个数列第三个数字数量；一个数字在当前块，可以用前面块的出现次数与后面块的出现次数 FFT 卷积得到前后和为某个数的出现次数，枚举当前块内数字统计即可。
时空复杂度	时间： $O(N^2/T + TA_i * \log(A_i))$ 空间： $O(A_i N/T)$

试题编号	Codechef Nov 12
名称	Martial Arts
题目大意	一个二分图，边有两个权值 $A_{i,j}$ 和 $B_{i,j}$ ，要进行匹配。令匹配边的 A 值总和为 H，B 值总和为 G。对手的目的是最大化 G-H，其次最大化 G，他会在知道匹配后选择是否去掉一条匹配边，使得该边的权值不算入 H 和 G。任务是找一个完全匹配，最大化 H-G，其次最大化 H。点数在 100 以内。 $A_{i,j}, B_{i,j} \leq 10^{12}$ 。
算法讨论	定义每条边的价值为 $W_{i,j} = A_{i,j} - B_{i,j}$ 。我们的任务是最大化和，其次最大化 H，对手的任务是最小化和，其次最大化 H。枚举对手删哪条边，利用 KM 算法实现“加入一条边”并且“强制匹配这条边”和“更改一条边权”，每条边对应一个 pair。就可以解决。
时空复杂度	时间： $O(N^4)$ 空间： $O(N^2)$

试题编号	Codechef Sep 13
名称	Two Roads
题目大意	平面上有 N 个点，需要加两条直线，令 d 为某个点到任意一条的最短距离，最小化 $\sum d^2$ 。 $3 \leq N \leq 100, 0 \leq x_i, y_i \leq 1000$ 。
算法讨论	对于加的直线的两条角平分线把平面分成了四个区域，每条直线控制了一个区域。发现这两条角平分线一定是垂直的。一定存在最优解的一条角平分线经过了两个点，另一条角平分线经过了一个点。于是可以枚举所有经过两个点的直线，然后按照其他点到这条线上垂足位置从小到大枚举。求具体直线可以用直线拟合的算法解决。
时空复杂度	时间： $O(n^3 * \log(n))$ 空间： $O(n)$

试题编号	Codechef Apr 14
名称	Chef and Tree Game
题目大意	<p>有一颗 <math>N</math> 个节点的有根树 <math>G</math>，树上的边是黑色或者白色。两名玩家玩一个树形删边游戏：两名玩家轮流行动；大厨每一次删除一条还在树上的黑边，Ksen 每次删去一条还在树上的白边；当一条边被删去之后，所有与根节点不连通的边也会一起被删去；当一名玩家无法行动时，他就输了。两人都用最优策略，求大厨先手谁会赢，Ksen 先手谁会赢。<math>1 \leq T \leq 1000, 1 \leq N \leq 10^5, 1 \leq U_i, V_i \leq N, 0 \leq C_i \leq 1</math>。</p>
算法讨论	<p>对每个节点定义一个局面函数 <math>f</math>，可以使用递推的方式得到：叶子的函数值为 0。每一个节点的函数值是所有孩子贡献之和。对于一个孩子 <math>i</math>，如果连接它的是黑边，那么令 <math>a</math> 为 <math>f_i + a &gt; 1</math> 的最小正整数，它的贡献是 <math>\frac{f_i + a}{2^a - 1}</math>；如果连接的是白边，那么令 <math>a</math> 为 <math>f_i - a &lt; -1</math> 的最小正整数，它的贡献就是 <math>\frac{f_i - a}{2^a - 1}</math>。如果根节点的 <math>f</math> 值为 0，那么谁后手谁赢；如果是整数，那么无论如何都是厨师赢；否则无论如何都是 Ksen 赢。需要用平衡树维护高精度的 <math>f</math> 值。</p>
时空复杂度	时间： $O(n * \log^2(n))$ 空间： $O(n * \log(n))$

试题编号	Codechef Jan 13
名称	A New Door
题目大意	一个 $W \times H$ 的长方形，初始都是白色的，有 $N$ 个圆，将这 $N$ 个圆覆盖的区域染黑。求最后白色范围的周长（不含长方形的四条边）。 $1 \leq T, W, H, N, X, Y \leq 1000, 1 < R \leq 1000$ 。X,Y,R 均恰好保留两位小数。所有圆都是不同的，要么圆心不同，要么半径不同。
算法讨论	题目中所求的长度等价于所有圆没有被其他圆覆盖的圆弧长度的和。这个可以 $N^2$ 求出每个圆被覆盖了哪些段（用极角区间表示）。然后转化为一个线段覆盖问题，可以贪心解决。可以把所有输入 $\times 100$ 转化为整数减少精度误差。
时空复杂度	时间： $O(N^2 * \log(N))$ 空间： $O(N^2)$

试题编号	Codechef May 14
名称	Dynamic Trees and Queries
题目大意	<p>给定一个有 <math>N</math> 个点的有根树，你需要执行以下 <math>M</math> 个询问。每个节点有一个 <math>key</math> 和一个 <math>value</math>，每个节点以他的 <math>key</math> 引用。<math>N</math> 个节点的 <math>key</math> 分别是 <math>0</math> 到 <math>N-1</math>。根节点的 <math>key</math> 总是 <math>0</math>。询问有以下 4 种类型。1. 给定一个节点的 <math>key</math>，生成一个新节点，作为这个节点的孩子，新节点的 <math>key</math> 将是从未使用的正整数中最小的一个，<math>value</math> 将给定。2. 给定一个节点的 <math>key</math>（称之为 <math>A</math>），将以 <math>A</math> 为根的子树中所有节点的 <math>value</math> 加上给定值。3. 给定一个节点的 <math>key</math>（称之为 <math>A</math>），将以 <math>A</math> 为根的子树中所有节点删除。注意删节点不释放 <math>key</math>。4. 给定一个节点的 <math>key</math>（称之为 <math>A</math>），询问以 <math>A</math> 为根的子树中所有节点的 <math>value</math> 值得和。强制在线。<math>n, m \leq 5 * 10^4</math>。</p>
算法讨论	<p>可以考虑用平衡树维护树的 <math>2n</math> 长度的 <math>dfs</math> 序。每个点出现两次的区间内表示的是这个点的子树。插入一个节点可以直接插在 <math>dfs</math> 序中 <math>A</math> 第一次出现的位置后面。子树删除、子树权值加和询问子树权值和，都可以利用平衡树解决。</p>
时空复杂度	时间： $O(M * \log(N) + N)$ 空间： $O(N)$

试题编号	Codechef May 12
名称	Little Elephant and Boxes
题目大意	有 $n$ 个盒子，每个盒子里有一些钱或者钻石。不知道每个盒子里具体是什么，但知道如果打开第 $i$ 个盒子，他有 $\frac{P_i}{100}$ 的概率能获得 $V_i$ 美元的钱，而有 $1 - \frac{P_i}{100}$ 的概率能获得一个钻石。现在有 $m$ 个物品，分别编号为 $0$ 到 $m-1$ ，第 $j$ 个物品需要花费恰好 $C_j$ 美元的钱和 $D_j$ 个钻石。小象非常聪明，当他获得了一定量的钱和钻石后，他总会买尽可能多的物品。每个物品只能购买一次。请计算出，当他打开所有的盒子后，他期望能够买到的物品个数是多少？ $1 \leq T \leq 5, 2 \leq n \leq 30, 1 \leq m \leq 30, 1 \leq V_i, C_j \leq 10^7, 0 \leq D_j \leq 30, 0 \leq P_i \leq 100$ 。
算法讨论	因为物品数和总钻石数比较少，首先背包预处理出有 $i$ 个钻石，买 $j$ 个物品需要多少钱。记作 $w_{i,j}$ 。考虑 meet in the middle，搜索前半部分的所有情况，得到一些三元组（钻石个数，钱的个数，概率），然后将这些三元组按照钻石个数为第一关键字，钱多少为第二关键字排序。去搜后面的部分，也得到一些三元组，这时可以枚举前面拿了多少钻石和最终获得了多少物品，这时可以根据背包得到需要的钱是在一个区间内的，然后二分前面的三元组，得到概率，答案对这些求和即可。
时空复杂度	时间： $O(n * m * \log(2^{\frac{2n}{3}}) * 2^{\frac{n}{3}} + 2^{\frac{2n}{3}})$ 空间： $O(2^{\frac{2n}{3}} + nm)$

试题编号	Codechef Aug 11
名称	Shortest Circuit Evaluation
题目大意	在短路计算的前提下，有些包含 and, or, not 操作的表达式，并且对于每一个变量都有为 true 的概率，你需要找到一个计算的顺序，使得在过程中期望的计算次数最小（指期望下使用的变量个数最少）。 $1 \leq t \leq 50, 0 < p < 1$ 。
算法讨论	记 $P$ 为正确的概率， $E$ 为就算次数的期望。可以推导出 and 连接的变量 $A_i$ 按照 $\frac{E_i}{1-P_{A_i}}$ 排序最优。or 连接的变量可以通过加 not 变成 and 连接的变量，于是套一个表达式求值可以解决。
时空复杂度	时间： $O(n * \log(n))$ 空间： $O(n)$

试题编号	Codechef Jul 11
名称	Billboards
题目大意	有 $N$ 个位置，选择一些位置放上物品，要求任意连续的 $M$ 个位置中至少有 $K$ 个位置必须有物品。求放物品数目最少的方案数。 $1 \leq T \leq 300, 1 \leq K \leq M \leq 500, M \leq N \leq 10^9$ 。
算法讨论	当 $m n$ 时，把整个分成 $n/m$ 块，每块中至少有 $k$ 个位置放物品。构造一个 $k * \frac{n}{m}$ 的矩阵 $A$ ，矩阵中第 $i$ 行第 $j$ 列表示第 $j$ 个块里第 $i$ 个物品的位置。如果这个矩阵是合法的，需要满足这个矩阵的每一行单调不增，每一列单调增。可以用杨氏图表的钩子公式解决。当 $m$ 不整除 $n$ 时，如果 $n \bmod m \leq m - k$ 那么每一组前 $n \bmod m$ 个位置都不放物品，否则每一组的后 $m - n \bmod m$ 个位置都放物品。去掉以后转化成整除的情况。
时空复杂度	时间： $O(MK + NK)$ 空间： $O(1)$

试题编号	Codechef May 14
名称	Sereja and Subsegment Increasings
题目大意	有一个包含 $n$ 个整数的序列 $A_i$ 。还有一个包含 $n$ 个整数的序列 $B_i$ 。在一次操作中，你可以选择两个下表 $i$ 和 $j$ ( $1 \leq i \leq j \leq n$ )，然后把所有 $A_i$ 到 $A_j$ 之间的元素（包含），增加 1 之后模 4。求将 $A$ 数组转换成 $B$ ，至少要执行多少次操作。 $1 \leq T \leq 10, 0 \leq A_i, B_i \leq 3, n \leq 10^5$ 。
算法讨论	令 $C_i = A_i - B_i \bmod 4$ , $D_i = C_{i+1} - C_i$ 。把问题转化成了最开始有一个全为 0 的数组，每次可以选两个数 $i, j$ ( $0 \leq i < j \leq n$ )，把第 $i$ 位 +1，第 $j$ 位 -1，问最多多少次变成 $D$ 。这个如果不是 $\bmod 4$ ，答案是所有正值的和。 $\bmod 4$ 的时候可以贪心解决，先不管 -1,0,1，如果有 2 或者 3，找一下有没有 -3 或 -2，如果有把当前位置 -4，那个位置 +4。然后在用不 $\bmod 4$ 的算法可以解决。
时空复杂度	时间： $O(n)$ 空间： $O(n)$



试题编号	Codechef Feb 12
名称	Find a Subsequence
题目大意	给你一个长度为 $N$ 的数组 $A_0, A_1, \dots, A_{n-1}$ 和一个字符串 $B$ , $B$ 是 “12345” 的排列。你需要找到一个长度为 5 的 $A$ 的子序列, 该子序列中的元素互不相等, 并满足他们的相对大小和 $B$ 一样。 $T \leq 20, 5 \leq N \leq 1000, -10^9 \leq A_i \leq 10^9$
算法讨论	枚举第 2 个数和第 4 个数字的位置, 这样得到一些大小关系, 剩下的三个数字再其中两个数字肯定可以取该区间内的最值, 剩下的数字是查询一个区间存在不存在一个范围内的数字, 可以用二维前缀和查询即可, 具体那个数字可以二分判断。然后判断是否合法即可。
时空复杂度	时间: $O(N^2)$ 空间: $O(N^2)$

试题编号	Codechef Apr 15
名称	Black-white Board Game
题目大意	初始时, 他们有一个 $N \times N$ 的矩阵, 矩阵的格子都是白色的。在游戏开始之前, Alex 会以如下方式给矩阵染色: 他选择 $N$ 对整数 $L[i], R[i]$ , 然后将第 $i$ 行的第 $L[i]$ 列到第 $R[i]$ 列 (从 1 开始计数) 染成黑色。玩家一轮轮进行游戏。在游戏中的每一轮, Alex 和 Fedya 都要行动一次, Alex 先行。当一位玩家行动时, 他需要给出整数 1 到 $N$ 的一个排列 $P$ , Alex 给出的排列中的逆序对数应为偶数 (Fedya 给出的为奇数), 另外, 所有 $(i, P_i)$ 对应的矩阵中的格子应为黑色。每轮游戏中玩家给出的排列不能与之前给出的重复。如果在同一轮中 Alex 和 Fedya 都无法给出满足条件的排列, 游戏和局。如果 Alex 和 Fedya 都找到了满足条件的排列, 游戏继续, 进入下一轮。如果其中一位玩家给出了满足条件的排列而另一位没有, 则给出排列的一方获胜, 游戏结束。双方都会采取最优策略进行游戏。求最后谁会赢。 $1 \leq L_i \leq R_i \leq N \leq 10^5$ 。
算法讨论	答案等价于矩阵的行列式是大于 0、等于 0 还是小于 0。普通的 Gauss 消元复杂度较高。注意到这个矩阵是每行一段连续的 1, 可以一列一列消, 对于第 $i$ 列, 用 $L=i$ , $R$ 最小的行去消其他的 $L=i$ 的行, 消完的行变成了 $L=R+1$ 的行, 直到消完。可以用线段树合并维护这个过程。
时空复杂度	时间: $O(N * \log(N))$ 空间: $O(N)$

试题编号	Codechef Apr 12
名称	Substrings on a Tree
题目大意	给一棵有根树，每个节点上有一个字母。询问从某个节点的祖先到这个节点构成的本质不同的字符串数量，和规定字典序的第 $K$ 大的字符串。 $1 \leq N \leq 250000, 1 \leq Q \leq 50000, 1 \leq K \leq 2^63 - 1$ 。最后的输出不超过 800KB,P 均为随机生成。
算法讨论	可以对树上所有节点到根的字符串建一个广义后缀自动机。然后本质不同的字符串个数可以记忆化 dfs 一遍后缀自动机得到。求第 $K$ 大，可以按照字母顺序依次访问每个节点，与线段树上访问某个点类似。
时空复杂度	时间： $O(\min(26 * Q * N, 800000) + N)$ 空间： $O(N * 26)$

试题编号	Codechef Dec 13
名称	Query on a tree VI
题目大意	给定一棵 $n$ 个节点的树，每个节点有一个颜色 (黑/白)，初始都为黑。维护一种数据结构，支持下列操作：0 u: 询问有多少点与点 $u$ 连通。两个点 $u$ 和 $v$ 是连通的，当且仅当 $u$ 到 $v$ 最短路径上的所有点 (包括 $u$ 和 $v$ ) 颜色都相同。1 u: 切换点 $u$ 的颜色 (黑变白，白变黑)。 $1 \leq n, m \leq 10^5$ 。
算法讨论	$f[i][0/1]$ 表示 $i$ 这个点为黑或白时的答案。当修改一个点 $x$ 时，实际上需要把 $x$ 到祖先同色的点上的同色/不同色同时加/减一个数。可以用树链剖分 + 线段树维护。
时空复杂度	时间： $O(m * \log^2(n))$ 空间： $O(n)$

试题编号	Codechef Feb 15
名称	Devu and Locks
题目大意	求有多少 $N$ 位十进制数是 $P$ 的倍数且每位之和小于等于 $M$ ，允许前导 0，答案对 998244353 取模。 $1 \leq N \leq 10^9$ ，两种数据：一种 $1 \leq P \leq 50, 1 \leq M \leq 500$ ，另一种 $1 \leq P \leq 16, 1 \leq M \leq 15000$ 。
算法讨论	$F_{i,j,k}$ 表示 $i$ 位数， $\text{mod } P=j$ ，和为 $k$ 的方案数。有 $F_{2*i,j,k} = F_{i,a,b} * F_{i,j-a*10^i \text{ mod } P, k-b}$ 。可以倍增第一维，枚举第二维，卷积第三维。倍增 +FFT 解决。
时空复杂度	时间： $O(P^2 * M * \log(N) + M * \log(N) * \log(M))$ 空间： $O(P * M)$

试题编号	Codechef Dec 12
名称	Different Trips
题目大意	一棵树，标号为 $1 \sim n$ ，其中城市 1 是根结点。假如两个节点的度数相同，则他们被认为是相似的。现在选择一些路径：首先确定节点 A，然后在 A 到根的路径上确定一个节点 B，然后城市 A 到 B 就是一条可行的路径。两条路径被认为是相似的当且仅当他们的长度相同且按顺序一一对应的节点都相似。求有多少种不相似的可行的路径。 $1 \leq n \leq 10^5$ 。
算法讨论	每个点的权值是每个点的度数，然后对这棵树当作 trie 树建出广义后缀自动机。然后就是一个后缀自动机上统计本质不同子串个数的问题。
时空复杂度	时间： $O(n)$ 空间： $O(n)$

试题编号	Codechef Mar 15
名称	Counting on a Tree
题目大意	给定一个包含 $N$ 个节点的有标号有边权无根树，点从一开始标号。你的任务是计算有多少无序数对 $(S,T)$ ，满足连接 $S,T$ 两点的路径上，所有边权的最大公约数等于 1。当然，我们只计算 $S \neq T$ 的数对。给定 $Q$ 组询问，其中第 $i$ 组询问形如 $A_i, C_i$ ，表示第 $A_i$ 条边的权值修改为 $C_i$ 。对于每一组询问，输出改动后对应的答案。 $1 \leq X, Y \leq N, X \neq Y, 1 \leq Z \leq 10^6, 0 \leq Q \leq 100, 1 \leq A_i \leq N-1, 1 \leq C_i \leq 10^6, 1 \leq N \leq 10^5$ 。
算法讨论	根据容斥原理，如果令 $f_i$ 为权值为 $i$ 的倍数的路径条数，那么答案就是 $\sum_{i=1}^Z f_i * \mu_i$ 。考虑离线，假设现在求 $f_i$ ，可以只考虑所有边权为 $i$ 的倍数的边，如果没有修过，就可以通过并查集来求出只考虑这些边的图中的路径条数。如果有修改那么可以先将于修过无关的边插入到并查集中，然后修过过程的边用可以后退的并查集维护。
时空复杂度	时间： $O(Z + (n + Q^2) * \log(n))$ 空间： $O(n)$

试题编号	Codechef Apr 12
名称	Find a special connected block
题目大意	给出一个 $N \times M$ 的矩阵，每个格子上填着从 -1 到 $N \times M$ 中的一个数。你的任务是找到一个联通块 (只能通过上下左右四个方向连接)，在这个联通块里，至少要包含 $K$ 个不同的正数，且不能有 -1。我们还会给出选取每一个格子所需要的代价，我们要求你选出的联通块所包含的格子的代价和最小。请输出这个最小值。 $1 \leq N, M \leq 15, 1 \leq K \leq 7$ 。
算法讨论	如果每一个格子的权值范围在 $[-1, k]$ 内，就可以用斯坦纳树解决。我们多次给 $[1, n \times m]$ 每个点随机一个 $[1, k]$ 的权值进行求解。每次只要随机的过程正确区分了最优解的 $K$ 个权值就可以得到最优解，单次正确率是 $\frac{k!}{k^k}$ 。随机 $c$ 次正确率是 $1 - \left(\frac{k^k - k!}{k^k}\right)^c$ 。
时空复杂度	时间： $O(c * n * m * 3^k)$ 空间： $O(n * m)$

试题编号	Codechef May 12
名称	Selling Tickets
题目大意	有 $n$ 道菜， $m$ 个人，这些人中有一些人去吃晚餐，晚餐时每个菜可以分配给一个人，如果第 $i$ 个人吃到了第 $a_i$ 道菜或第 $b_i$ 道菜，就开心了。现在求最大的 $x$ ，使得任意 $x$ 个人来吃饭都开心。 $1 \leq n \leq 200, 1 \leq m \leq 500$ 。
算法讨论	问题等价于给一个 $n$ 个点 $m$ 条边的无向图，问边数恰好比点数多 1 的连通子图的点数最少是多少。发现会有两种情况：1. 两个点之间的三条路径。这种情况可以枚举两个点然后 bfs 判断。2. 两个由一条路径连接起来的简单环。此时可以枚举其中一个度数至少为 3 的点，以它为根求 <i>BFS</i> 树，此时每一个简单环代表了树上的一条路径且每一个点到树根的最短路径都是树边，我们把每一个环的权值设为它的边数加上它两端的 LCA 到根的距离，只需要拿此时权值最小的两个环的权值之和更新答案就行了。
时空复杂度	时间： $O(n^2 m)$ 空间： $O(n + m)$

试题编号	Codechef Feb 14
名称	Graph Challenge
题目大意	给定一个 $n$ 个点 $m$ 条边的有向图的 dfs 序，保证所有节点都能从 1 号点到达。一个节点 $x$ 对 $y$ 来说是好的当且仅当存在一条 $x$ 到 $y$ 的路径使得中间节点编号都大于 $y$ 。一个节点 $x$ 对 $y$ 来说是好的当且仅当它是所有对 $y$ 的好节点中编号最小的。给定 $Q$ 个询问，对于每个询问，求对于某个节点它是多少个节点的好节点。 $1 \leq T \leq 10, 1 \leq n, Q \leq 10^5, n-1 \leq m \leq 2*10^5$ 。
算法讨论	他的定义正好是 Dominator tree 中的半必经点，按照它的 dfs 序求半必经点即可。
时空复杂度	时间： $O(n\alpha_n)$ 空间： $O(n)$

试题编号	Codechef Aug 14
名称	Team Sigma and Fibonacci
题目大意	<p>给两个数字 <math>M, N</math>，求</p> $\left( \sum_{x+y+z=N} 6 * x * y * z * fib_x * fib_y * fib_z \right) \bmod M$ <p>。其中 <math>fib_i</math> 等于斐波那契数列第 <math>i</math> 项。<math>0 \leq N \leq 10^{18}, 1 \leq M \leq 10^5, \sum M \leq 10^6</math>。</p>
算法讨论	<p>可以推出这个式子的母函数是 <math>f(x) = 6g(x)x^3(x^2+1)^3</math>，其中</p> $[x^n]g(x) = \frac{1}{3125} (25 \binom{n+5}{5} (fib_{n+6} + 2fib_{n+5})$ $+ 150 \binom{n+4}{4} fib_{n+5} + 5 \binom{7}{2} \binom{n+3}{3} (fib_{n+4} + 2fib_{n+3})$ $+ 5 \binom{8}{3} \binom{n+2}{2} fib_{n+3} + \binom{9}{4} (n+1) (fib_{n+2} + 2fib_{n+1}) + \binom{10}{5} fib_{n+1})$ <p>求 <math>fib_i</math> 用矩阵乘法，组合数都不超过 5 项。参考：  <a href="http://www.mathstat.dal.ca/FQ/Scanned/15-2/hoggatt1.pdf">http://www.mathstat.dal.ca/FQ/Scanned/15-2/hoggatt1.pdf</a>  和 <a href="http://pan.baidu.com/s/1sjqGiDn">http://pan.baidu.com/s/1sjqGiDn</a></p>
时空复杂度	时间： $O(\log(n))$ 空间： $O(1)$

试题编号	Codechef Jun 15
名称	Chefbook
题目大意	<p>有 <math>n</math> 个人，每个人是一个或多个人的好友，好友关系是单向的，每一个人 <math>x</math> 对他的每个好友都有一个喜爱值 <math>L[x][y]</math>。你需要给出 <math>P[x]</math> 和 <math>Q[y]</math>，使得 <math>W[x][y] = L[x][y] + P[x] - Q[y]</math>，每对 <math>x</math> 和 <math>y</math> 的喜爱值 <math>W[x][y]</math> 都在给定的下界 <math>S[x][y]</math> 与上界 <math>T[x][y]</math> 之间（包括 <math>S[x][y]</math> 与 <math>T[x][y]</math>）。让 <math>W[x][y]</math> 的和尽可能大。<math>1 \leq T \leq 3, 1 \leq N \leq 100, 1 \leq x, y \leq N, -600 \leq L[x][y] \leq 600, -1000 \leq S[x][y] \leq T[x][y] \leq 1000, 0 \leq P[x], Q[y] \leq 1000000</math>。</p>
算法讨论	<p>可以把每个约束拆成两个不等约束条件，然后得到的线性规划的矩阵是全幺模的，可以对偶。对偶后就可以用最小费用最大流求解了。然后利用费用流剩余网络，用差分约束问题求出一组具体解。</p>
时空复杂度	时间： $O(NM^2)$ 空间： $O(N + M)$

试题编号	Codechef Apr 15
名称	Little Party
题目大意	<p>Leha 有 <math>N</math> 个朋友，他之前已经开过 <math>M</math> 场派对。由于某些原因，这些派对都不太愉快。他所有的朋友无论情绪如何，他们都会前来赴约。每次他的朋友中都会有一些子集导致正常派对不欢而散。现在 Leha 想要分析先前的派对，找到他的朋友中在某种情绪状态下容易出问题的群体 (可能为空)。他清楚地记得之前每次聚会时每个人的情绪状态。每个朋友可能心情好 (用大写字母表示) 或者心情不好 (用小写字母表示)。相同的字母 (无论大小写) 表示同一个朋友，不同的字母表示不同朋友。如果一个子集中只要存在某个元素就会带来问题，并且全部已知，如果都不存在就不会带来问题，叫做基子集。求长度和最小的基子集。<math>1 \leq T \leq 120, 0 \leq M \leq 1000, 1 \leq N \leq 5</math>。</p>
算法讨论	<p>题目描述的是一个最小带权集合覆盖的问题，直接搜索复杂度较高，可以简化状态，如果 <math>s_1, s_2</math> 选了，那么 <math>s_1 s_2</math> 就没有必要选。这样处理出来合法状态最多有 32 个基子集，再加上最优性剪枝。就可以出解了。</p>
时空复杂度	时间： $O(2^{32})$ 空间： $O(n)$

试题编号	Codechef Feb 14
名称	Count on a Treap
题目大意	在计算机科学中，Treap 是一种常见的数据结构。 <a href="http://en.wikipedia.org/wiki/Treap">http://en.wikipedia.org/wiki/Treap</a> 这道题目要求你维护一个大根堆 Treap，要求支持：0 k w：插入一个关键字为 k，权值为 w 的点 1 k：删除一个关键字为 k 的点。2 ku kv：返回关键字分别为 ku 和 kv 两个节点的距离保证任意时刻树中结点 key 和 weight 都是两两不同的。不会删除当前 Treap 中不存在的点 $1 \leq N \leq 2 * 10^5, 0 \leq k, w, ku, kv \leq 2^{32}$ 。
算法讨论	考虑维护 treap 的 dfs 序，两点的 lca 就是对应区间 val 最大的点。所以只需能求每个点的深度即可，发现每个点的深度是两个方向的单调序列的长度和。可以每次用线段树 $\log(n)$ 实现。然后查询距离就是查询两点的 lca 和三个点的深度。
时空复杂度	时间： $O(n * \log^2(n))$ 空间： $O(n)$

试题编号	Codechef Dec 14
名称	Course Selection
题目大意	课业计划共包含 N 项课程，每项课程都需要在 M 个学期里的某一个完成。一些课程有前置课程：对于所有的 $i(1 \leq i \leq K), A[i]$ 是 $B[i]$ 的前置课程。相同的课程在不同的学期中可能会由不同的教授授课，不同的教授可能会影响铃在这一门课程上的表现。我们给出数组 X 来描述这一信息。对于每项课程 i 和学期 j。 $X[i][j]$ 表示铃在学期 j 选修课程 i 所能得到的期望分数。如果 $X[i][j] = -1$ ，则表示那个学期没有这个课程。求所能得到期望分数的平均值的最大值。 $1 \leq M, N \leq 100, 0 \leq K \leq 100, -1 \leq X[i][j] \leq 100, 1 \leq A[i], B[i] \leq N$ ，数据保证至少有一组解。
算法讨论	考虑网络流建图。可以对每个课程按学期拆点，连成一串，然后这些串之间连边就能起到限制时间的作用。连的每条边的权值是最大值 - 这门课当前的值。这样求一个最小割就能得到一个最大和。除以总数就是最大平均值。
时空复杂度	时间： $O(Maxflow(M * N, M * N))$ 空间： $O(M * N)$



试题编号	Codechef Jul 11
名称	Trial of Doom
题目大意	约翰尼进入了一个大房间，这个房间被划分成了 $n*m$ 个格子，每个格子可以是蓝或红。现在约翰尼在 $(1,1)$ ，出口在 $(n,m)$ ，他需要到达终点并使得所有格子都是蓝色的。约翰尼可以移动到八个相邻的格子上，每当他离开一个格子，那么这个格子和它周围的四个格子会改变颜色。现在给出房间的颜色情况，约翰尼想知道他是否能离开这个房间。 $1 \leq t \leq 50, 1 \leq n, m \leq 10^9, \min(n, m) < 40, 0 \leq k \leq \min(m * n, 10000), 1 \leq x \leq n, 1 \leq y \leq m$ 。
算法讨论	可以构造出在一个 $2*2$ 的方格中，在不改变方格状态的情况下，可以从当前方格移动到任意一个方格。当 $\min(n, m) > 1, n \leq m$ 时，对于 $(x, y)$ 如果是红色可以操作 $(x-1, y)$ 变成蓝色，每个方格代表了第一列方格的状态，同一行中会出现循环，可以用高斯消元或线性基求最后一列的状态。当 $n=1$ 时，先不考虑来回走，那么只能从 $(1,1)$ 走到 $(1,m)$ 。来回走和返回的步数和一定是偶数，计算出步数判断是否为偶数即可。
时空复杂度	时间： $O(k + \min(n, m) * Len)$ 空间： $O(\min(n, m) * Len)$

试题编号	Codechef Dec 13
名称	Petya and Sequence
题目大意	给出一个长度为 $n$ 的序列 $A[0..n-1]$ ，问是否存在一个长度为 $n$ 的序列 $B[0..n-1]$ 满足：-至少存在一个 $0 \leq i \leq n-1$ 满足 $B[i] \neq 0$ -对于任意 $0 \leq j \leq n-1$ 满足 $[0 \leq i \leq n-1] A[i] * B[(i+j) \bmod n] = 0$ $T$ 组数据。 $1 \leq T \leq 100, 1 \leq n \leq 3 * 10^4, -1000 \leq A[i] \leq 1000, \sum n \leq 1.5 * 10^5$ 。
算法讨论	问题等价于判断矩阵 $X(X_{i,j} = A[(i+j) \bmod n])$ 是否满秩。这个矩阵的秩等于 $n - \deg(\gcd(f(x), x^n - 1))$ 。然后根据分圆多项式，求这个式子可以，枚举 $n$ 的因数 $d$ ，令 $b_i = \sum_{j \bmod d = i} a_j$ ，枚举 $d$ 的质因子 $p$ ，令 $b'_i = b_{(i - \frac{d}{p}) \bmod d} - b_i$ ，然后令 $b = b'$ 。如果操作结束后对于任意 $0 \leq i < d$ 满足 $b_i = 0$ ，说明存在，否则枚举下一个因子 $d$ 。如果所有因子都不存在，就是不存在。
时空复杂度	时间： $O(5 * n * 96)$ 空间： $O(n)$

试题编号	Codechef Feb 12
名称	Flight Distance
题目大意	有 $N$ 个城市标号从 0 到 $N-1$ ，并且在这些城市间有 $M$ 条直达航线。每条航线可以用两个城市的编号和距离来表示（航线是双向的）。厨师将第 $i$ 条航线的距离（增加或减少）改变 $d_i$ （ $d_i$ 为有理数，改变之后航线之间的距离依然为正数），并付出 $d_i$ 的金额。使得在改变之后，没有一条航线的距离比经过中转点的距离还要长。厨师需要求出最少花费的金额数。输出分数。 $1 \leq N \leq 10, 1 \leq M \leq C(N, 2), 1 \leq D_i \leq 20$ 。
算法讨论	对于每条边建立两个变量 $d_i^+, d_i^-$ ，设 $g_{i,j}$ 为 $i$ 和 $j$ 之间的最短路，那么 $\forall x_k = i$ ，有 $g_{i,j} \leq W_k + d_k^+ - d_k^- + g_{y_k,j}$ ，相似的， $\forall y_k = i$ ，有 $g_{i,j} \leq W_k + d_k^+ - d_k^- + g_{x_k,j}$ 。特别的 $g_{i,i} = 0$ ，还需保证每条边长度大于 0，需要最小化的是 $\sum_{i=1}^M d_i^+ + d_i^-$ ，用小数跑单纯形，枚举分母转成分数即可。
时空复杂度	时间： $O(\text{单纯形})$ 空间： $O(N^2)$

试题编号	Codechef Nov 13
名称	Gangsters of Treeland
题目大意	一棵 $n$ 个点的有根树，根为 0。初始每个点都有一个颜色。在树上移动如果两个点颜色不同，需要付出 1 单位代价。有一些操作把根到某个点 $u$ 上的所有点染成另一种颜色。询问给定一个点 $u$ ，求 $f(u)$ ，定义 $f(u)$ 为以 $u$ 为根的子树中所有节点到根节点的代价的平均值。 $1 \leq T \leq 15, 1 \leq N \leq 10^5, 1 \leq Q \leq 10^5, \sum N \leq 2 * 10^5, \sum Q \leq 2 * 10^5$ 。
算法讨论	如果知道子树大小，只需求所有点到根的代价和。每次染色的操作很像 LCT 中的 access 操作。所以可以用一个 LCT 维护树的颜色。然后代价和可以用树状数组维护 dfs 序解决。
时空复杂度	时间： $O(n * \log^2(n))$ 空间： $O(n)$

试题编号	Codechef May 13
名称	Queries on tree again!
题目大意	给定一棵 $n$ 个点 $n$ 条边的基环树。模拟一下两种操作：f u v: 对 $u$ 到 $v$ 的最短路径上的所有边的权值取相反数。? u v: 在 $u$ 到 $v$ 的最短路径上，找到一个连续的边的集合，使得集合中边的权值之和最大。 $1 \leq N \leq 10^5, 1 \leq u, v \leq N, -10000 \leq c \leq 10000, 1 \leq Q \leq 10^5$ 。
算法讨论	对环上每个树维护一个树链剖分，最后再换上维护一棵线段树，每次操作是换上的一段。 $O(\log(n))$ 次线段树操作。
时空复杂度	时间: $O(n * \log^2(n))$ 空间: $O(n)$

试题编号	Codechef Dec 11
名称	Short II
题目大意	给定 $p$ (一个质数)，问有多少对 $a, b (a > p, b > p)$ 满足 $ab$ 被 $(a-p)(b-p)$ 整除。 $1 < p < 10^{12}, 1 \leq t \leq 5$ 。
算法讨论	题目的整除条件可以变为 $ab   p(a+b+p)$ 而不改变答案，考虑三种情况：1. $p$ 同时整除 $a$ 和 $b$ ，这时候的 $a, b$ 有 $(1,1), (1,2), (2,1), (2,3), (3,2)$ 。2. $p$ 不整除 $a$ 且不整除 $b$ ，有 $a \neq b$ ，不妨设 $a < b$ ，满足 $a < w = 1 + \sqrt{p+1}$ 。如果能够找到一个合适的 $a$ ，可以判断出 $b = \frac{a+p}{ak-1}$ ，令 $d = ak-1$ ，可以通过遍历 $d \leq \sqrt{p+w}$ 或遍历 $b \leq \sqrt{p+w}$ 可以在 $O(\sqrt{p})$ 的时间复杂度找到所有的 $a, b$ 。3. $ab$ 中恰好有一个被 $p$ 整除，可以发现这种条件的对数是 2 的两倍，对于 2 的数对 $(a, b)$ 可以生成 $(a, \frac{p*(a+p)}{b})$ ，和 $(\frac{p*(b+p)}{a}, b)$ 。
时空复杂度	时间: $O(T * \sqrt{p})$ 空间: $O(1)$

试题编号	Codechef Aug 11
名称	Something About Divisors
题目大意	对于给定的正整数 B 和 X，求满足条件的正整数 N 的个数：要求对于 N，至少存在一个数 D ( $N < D \leq B$ ) 能整除 $N * X$ 。 $T \leq 40, B \leq 10^{12}, X \leq 60$ 。
算法讨论	令正整数 $i = \frac{NX}{D}$ ，显然有 $i < X$ ，考虑枚举 i，避免重复可以计算满足 $i   NX$ 且不存在 j 满足 $i < j < X$ 且 $j   NX$ 的 N。因为 $i   NX$ ，所以有 $\frac{i}{\gcd(i, X)}   N$ ，那么 $N = A_i p$ ，因为有 $N \leq \frac{B_i}{X}$ ，所以有 $p \leq \frac{B_i}{XA_i}$ ，把 maxp 记作 P。因为 $j   A_i X * p$ ，所以有 $\frac{j}{\gcd(A_i X, j)}   M$ ，令 $B_j = \frac{j}{\gcd(A_i X, j)}$ 。那么满足 $i   NX$ 的同时满足 $j   NX$ 的条件是 $D_j   M$ ，所以可以枚举 i 对于 $i < j < X$ 容斥， $ans_i = \sum (-1)^t \frac{P}{lcm(B_i)}$ 。可以用 DP 的方式进行。
时空复杂度	时间： $O(X^4)$ 空间： $O(X^3)$

试题编号	Codechef Dec 14
名称	Divide or die
题目大意	平面上给一个 N 度角，请你尺规作图 N 等分这个角。 $0 < n < 360$ ，操作次数 $< 1000$ 。
算法讨论	可以发现 $3   n$ 时无解，其他情况都有解。可以先画一个五边形，得到 72 度角，然后画一个三角形，得到 60 度角，然后作差得到 12 度角，平分两次得到 3 度角。然后用原角 mod 3 度角，得到 1 度或 2 度，2 度再平分下，这样有了一度角，可以 n 等分原来角了。
时空复杂度	时间： $O(n)$ 空间： $O(n)$

试题编号	Codechef Feb 15
名称	Payton numbers
题目大意	<p>定义三元组的乘法 <math>\text{def multiply}((a1,b1,c1), (a2,b2,c2))</math>: <math>s = (a1a2 + b1b2 + c1c2) + (a1b2 + b1a2) + (c1 + c2)</math> <math>t = \text{floor}[s/2] + 16(c1 + c2) - c1c2</math> <math>A = (t - 2(a1b2 + b1a2) - (a1c2 + c1a2) + 33(a1 + a2) + (b1b2 - a1a2))</math> <math>B = (t - 5(a1b2 + b1a2) - (c1b2 + b1c2) + 33(b1 + b2) + (2b1b2 + 4a1a2))</math> if <math>s</math> is even: return <math>(A-540,B-540,24)</math> else: return <math>(A-533,B-533,11)</math></p> <p>定义 zero: 若 <math>x * \text{任何 } y = 0</math>, 则称 <math>x</math> 是 zero 定义单位元, 若 <math>x * \text{任何 } y = y</math>, 则称 <math>x</math> 是单位元定义质数, 若 <math>x</math> 不是 zero 且不能分解成两个非单位元的乘积, 则称 <math>x</math> 是质数给定一个三元组, 问他是不是质数 <math>T \leq 10^4, c = 11 \text{ 或 } 24, -10^7 \leq a, b \leq 10^7</math>。</p>
算法讨论	<p>令 <math>w</math> 为满足方程 <math>w^2 = w - 3</math> 的解, 即 <math>w = \frac{1+\sqrt{-11}}{2}</math>, 那么对于每个三元组, 都有到域 <math>\mathbb{Z}[w]</math> 的映射, <math>\phi(a, b, c) = (33 - 2a - c) + (b - a)w</math>。于是判断 <math>a + bw</math> 是否为素数。令 <math>Nx = x(x.a + x.b - x.bw)</math>。有结论 (1) 如果 <math>x</math> 不是整数, 那么 <math>x</math> 是质数, 当且仅当 <math>Nx</math> 是质数。(2) 如果 <math>x</math> 是整数, 那么 <math>x</math> 是质数当且仅当 <math>x</math> 是质数且要么 <math> x =2</math>, 要么 <math> x  \neq 11</math> 且 <math>-11</math> 在模 <math>x</math> 域下没有二次剩余。用 Cipolla's algorithm 和 Miller Rabin 判断即可。</p>
时空复杂度	时间: $O(T * \log( a ))$ 空间: $O(1)$

试题编号	Codechef Aug 15
名称	Simple Queries
题目大意	<p>给定一个含 <math>N</math> 个正整数的数组 <math>A</math>。现有关于它的 <math>Q</math> 个询问，询问有以下五种类型：</p> <p>1 l r : 令 <math>S</math> 为由下标范围从 <math>l</math> 到 <math>r</math> 的不同的元素构成的有序集合。你要求出</p> $\left( \sum_{1 \leq i < j < k \leq  S } S_i S_j S_k \right) \bmod (10^9 + 7)$ <p>。</p> <p>2 x y : 将下标为 <math>x</math> 的元素赋值 <math>y</math>      3 x : 将下标为 <math>x</math> 的元素从数组中删除</p> <p>4 z y : 在下标为 <math>z</math> 的元素之后插入元素 <math>y</math>，若 <math>z</math> 等于 0，则在数组最前端插入</p> <p>5 l r : 输出下标在 <math>l</math> 到 <math>r</math> 范围内的不同元素个数      数组下标从 1 开始。数据保证数组总是非空。<math>1 \leq N, Q \leq 10^5, 1 \leq A_i, y \leq 10^9 + 6, 1 \leq x \leq  A , 0 \leq z \leq  A , 1 \leq l \leq r \leq  A </math>。</p>
算法讨论	<p>把一个位置记为 <math>(x, y)</math>，其中 <math>x</math> 为它的下标，<math>y</math> 为它后面第一个数字和它相同的位置的下标，可以用平衡树预处理，然后每个插入操作就可以避免下标的编化。可以用树状数组维护，对于询问操作只需询问 <math>x \in [L, R], y &gt; R</math> 的点的信息。需要维护 1 次方和，2 次方和，3 次方和。可以计算出答案。修改可以用一个 set 来维护出现的下标。然后更新到树状数组上。</p>
时空复杂度	时间: $O(m * \log^2(n))$ 空间: $O(m * \log^2(n))$

试题编号	Codechef Jun 11
名称	Attack of the Clones
题目大意	<p>称一个形为 <math>f:A \rightarrow B</math> 的函数叫做布尔函数，其中 <math>A</math> 是所有长度为 <math>n</math> 且仅由 0 和 1 组成的数列的集合，<math>B=0,1</math>，我们称 <math>n</math> 为布尔函数的项数。我们称满足一些条件的布尔函数构成的集合称为 clone。现在有四个特殊的 clone 如下：</p> <p><math>Z</math> 是 0-保留值函数集合：满足 <math>f(0, \dots, 0) = 0</math>；<math>P</math> 是 1-保留值函数集合：满足 <math>f(1, \dots, 1) = 1</math>；<math>D</math> 是自对偶函数集合：满足 <math>f(x_1, \dots, x_n) = f(\neg x_1, \dots, \neg x_n)</math>；<math>A</math> 是仿射函数集合：满足如果 <math>f(a_1, \dots, c, \dots, a_n) = f(a_1, \dots, d, \dots, a_n)</math> 则 <math>f(b_1, \dots, c, \dots, b_n) = f(b_1, \dots, d, \dots, b_n)</math> 的函数，在这里 <math>c</math> 和 <math>d</math> 都在某个位置 <math>i</math>，并且这个对于任意 <math>i, a_1, \dots, a_n, b_1, \dots, b_n, c, d</math> 都应成立。现在我们有兴趣知道在上述几种集合的组合中有多少个 <math>n</math> 项函数。<math>1 \leq n, q \leq 100</math>。</p>
算法讨论	<p>对于任意函数，可以用长度为 4 的二进制来表示，每一位表示是否属于这一位代表的集合。可以先求出所有二进制表示的集合个数。接着可以用一个 16 位的二进制数来表示这 16 个部分是不是属于一个集合。可以把前面集合的运算变成位运算。然后用栈处理表达式求值即可。然后累加起来就得到答案。</p>
时空复杂度	时间： $O(T S )$ 空间： $O( S )$

试题编号	Codechef Mar 15
名称	Random Number Generator
题目大意	<p>给一个常系数线性齐次递推序列。给定初始序列的前 <math>k</math> 项，<math>A_1, A_2, \dots, A_k</math> 和 <math>k</math> 个系数 <math>C_1, C_2, \dots, C_k</math>，序列的任意一项也就确定了。给定 <math>N</math>，你的任务是输出 <math>A_N</math>。<math>1 \leq N \leq 10^{18}, 0 \leq A_i, C_i \leq 104857601, 1 \leq k \leq 30000</math>。</p>
算法讨论	<p>直接矩阵乘法是 <math>O(k^3 \log(n))</math> 的，可以发现求出这个矩阵的特征多项式为 <math>f(x) = x^k - \sum_{i=1}^k C_i x^{k-i}</math> 可以用特征多项式优化矩阵乘法。可以用 FFT + 快速幂 + 多项式除法取模得到答案。</p>
时空复杂度	时间： $O(k * \log(k) * \log(n))$ 空间： $O(k)$

试题编号	
名称	
题目大意	
算法讨论	
时空复杂度	时间： $O(1)$ 空间： $O(1)$