# 2016国家集训队作业试题泛做表格

安徽师范大学附属中学吴作凡

2016 年 1 月 14 日

Problem 1

| 试题名称 | A New Door | | |
|---|---|---|---|
| 试题来源 | Codechef JAN 13 | 试题编号 | ANDOOR |
| 题目大意 | | 算法讨论 | |
| 给一个矩形区域和n个圆，问这些圆的并在矩形内的周长。<br>数据范围:$n \leq 1000$ | | 可以对于每个圆分开考虑，求出每个圆周没被覆盖的长度，只需要用其他所有的圆以及矩形和这个圆求交，将角度区间排序后统计答案就可以了。<br>要注意精度问题。 | |
| 时间复杂度 | $O(n^2logn)$ | 空间复杂度 | $O(n)$ |

Problem 2

| 试题名称 | Dynamic Trees and Queries | | |
|---|---|---|---|
| 试题来源 | Codechef MAY 14 | 试题编号 | ANUDTQ |
| 题目大意 | | 算法讨论 | |
| 给一个n个点的带点权的数，有m个操作，操作有以下四种：<br>1.加入一个给定点权和父亲的点。<br>2.删除一个子树。<br>3.给一个子树所有节点权值都加上一个值。<br>4.询问一个子树的权值和。<br>数据范围:$n, m \leq 10^5$ | | 因为只有子树操作而没有链操作，最容易的算法就是用平衡树（我使用了splay）维护树的dfs序，每次加入点就在相应dfs序位置加入点，删除子树时直接删除就好了，支持区间加以及区间求和操作的平衡树还是非常容易的。 | |
| 时间复杂度 | $O(nlogn)$ | 空间复杂度 | $O(n)$ |

Problem 3

<table>
<tr><td>试题名称</td><td colspan="3">The Baking Business</td></tr>
<tr><td>试题来源</td><td>Codechef OCT 11</td><td>试题编号</td><td>BAKE</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">有n个操作，有增加订单或者询问两种操作。增加订单和询问的格式相当复杂，这里略去不谈，可以去看详细题面。<br>数据范围:$n \le 10^5$</td><td colspan="2">开一个七维数组统计一下订单数量，如果某一维为0则表示没有限制，增加订单就直接手工讨论，询问的时候枚举一下年龄就好了。</td></tr>
<tr><td>时间复杂度</td><td>$O(n)$</td><td>空间复杂度</td><td>$O(1)$</td></tr>
</table>

Problem 4

<table>
<tr><td>试题名称</td><td colspan="3">Billboards</td></tr>
<tr><td>试题来源</td><td>Codechef JULY 11</td><td>试题编号</td><td>BB</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">一个长度为n的01序列，它的任意长度为m的连续子序列中必须都要有恰好k个1，问满足该条件的序列中1的数目最少的不同序列个数，答案模$10^9+7$。<br>数据范围:$k, m \le 50, m \le n \le 10^9$</td><td colspan="2">先考虑m整除n的情况，可以把串分成$\frac{n}{m}$段，显然每段至少有k个1，可以把这k个1都放在该段最后，那么1的个数最少就是$\frac{nk}{m}$个。<br>那么可以对于一个序列构造一个$k \times \frac{n}{m}$的矩阵，第i行第j列表示第j段第i个1在该段中的位置，如果一个序列合法那么矩阵的每一行单调不增，每一列单调增。我们只要统计这样的矩阵个数。<br>这就是一个半标准的杨氏矩阵，答案的公式如下$\prod_{1 \le i \le n, 1 \le j \le m} \frac{r+j-i}{n+m-i-j+1}$，这里n是行数，m是列数，r是可选数字数量。但是矩阵太大不能直接计算，但是大部分分子和分母都可以约分，最后只有$O(mk)$个数字需要计算。<br>当m不整除n的时候，令$t \equiv n (mod\ m), 1 \le t \le m$，我们可以用以下性质将问题转化为m整除n的情况：<br>1.$t \le m-k$：每一组前t个数字均为0。<br>2.$t > m-k$：每一组后m-t个数字均为1。</td></tr>
<tr><td>时间复杂度</td><td>$O(mk)$</td><td>空间复杂度</td><td>$O(1)$</td></tr>
</table>

Problem 5

| 试题名称 | Union on Tree | | |
|---|---|---|---|
| 试题来源 | Codechef OCT 14 | 试题编号 | BTREE |
| 题目大意 | | 算法讨论 | |
| 给定一棵n个点的树，每条边长都为1，有Q组询问，第i次询问给定Ki个数对$(a_i, r_i)$，表示和点$a_i$距离在$r_i$以内的点都被守护了，询问共有多少点被守护。<br>数据范围:$n, Q \leq 5*10^4, \sum K \leq 5*10^5$ | | 先考虑一个弱化的问题：询问和点$a$距离在$r$以内的点数，我们令这种询问叫做$func(a, r)$。<br>这个问题我们可以使用点分治，记录下点分治的结构以及重心的每个子树到重心距离在k以内的点数。询问的时候依次统计每层结构点数。这样预处理的复杂度是$O(nlog^2n)$，每次询问的复杂度是$O(logn)$。<br>接下来考虑原问题，我们可以对这K个点建立虚树，若点x和y的距离是$d_{xy}$而$r_x - d_{xy} > r_y$，那么可以用$r_x - d_{xy}$来更新$r_y$，于是可以用类似最短路的算法更新每个点的守护距离。然后分每条边讨论，任意一条边$(x, y)$都可以找到一个点$z$使得$r_x - d_{xz} = r_y - d_{zy}$（为了使得z在整点一开始可以在每条边上多加一个节点），那么答案就是$\sum_{x \in V} func(x, r_x) - \sum_{(x,y) \in E} func(z, r_x - d_{xz})$。 | |
| 时间复杂度 | $O(nlog^2n + \sum Klogn)$ | 空间复杂度 | $O(nlogn)$ |

Problem 6

| 试题名称 | Black-white Board Game | | |
|---|---|---|---|
| 试题来源 | Codechef APRIL 15 | 试题编号 | BWGAME |
| 题目大意 | | 算法讨论 | |
| 有一个$n \times n$的矩阵A，第i行$L_i$到$R_i$是1，其余均为0。<br>两个人玩游戏，每次他们都要报出一个未出现排列P，满足$A_{ip_i} = 1$，第一个人的排列逆序对数是奇数，第二个人是偶数，报不出来就输了，如果同时报不出来就平局。<br>判断游戏结果。<br>数据范围:$n \leq 10^5$ | | 根据行列式的定义可以发现偶排列数量减去奇排列数量就是A矩阵的行列式值，那么我们就只要求出A的行列式的符号。<br>求行列式的值的方法主要是消元，可以发现在消元的过程中我们可以让每行的1都是连续一段，我们可以用n个左偏树来维护每行的连续段，如果第i行是从$L_i$到$R_i$就把$R_i$扔到第$L_i$个左偏树中。从小到大遍历每一列，在当前左偏树里找到$R$最小的位置记为$Rmin$，将$Rmin$放到第i行，删去$Rmin$，再把这个左偏树合并到$Rmin + 1$里去，就可以完成消元的过程了。<br>要注意行编号的变化。 | |
| 时间复杂度 | $O(nlogn)$ | 空间复杂度 | $O(n)$ |

Problem 7

| 试题名称 | Card Shuffle | | |
|---|---|---|---|
| 试题来源 | Codechef JAN 12 | 试题编号 | CARDSHUF |
| 题目大意 | | 算法讨论 | |
| 有n张卡片，从1到n从上到下摆放。m次操作，每次取前A张，再取前B张，把前A张放回去，再取前C张，把B张倒序放回，最后把C放回。询问最后顺序。<br>数据范围:$n, m \leq 10^5$ | | 使用可以打翻转标记的平衡树模拟即可。 | |
| 时间复杂度 | $O(mlogn)$ | 空间复杂度 | $O(n)$ |

Problem 8

| 试题名称 | Chef and Balanced Strings | | |
|---|---|---|---|
| 试题来源 | Codechef MAY 15 | 试题编号 | CBAL |
| 题目大意 | | 算法讨论 | |
| 一个字符串是平衡的当且仅当它的每一个字符都出现了偶数次，一个字符串的type权值是它的所有平衡子串长度的type次方的和。现在给定一个长度为n的由小写字符构成的字符串。Q组询问，给定L、R、type，询问这个字符串的从L到R的子串的type权值是多少。<br>数据范围:$n, q \leq 10^5, type \in \{0, 1, 2\}$ | | 用一个26位二进制数$A_i$表示1 i中每个字符出现次数的奇偶性，那么子串(L,R)平衡当且仅当$A_{L-1} = A_R$，先把A数组离散化。<br>可以发现一般的数据结构都不太容易维护，所以要用到分块。直接对于A数组分块，块大小为$n^{0.5}$，于是可以预处理前i块数字j的出现次数、位置和、位置平方和，第i块到第j块的答案，那么每次询问可以先找到中间一段整块的答案，再暴力两端不属于整块的部分就好了，可以使用时间戳。 | |
| 时间复杂度 | $O(n^{1.5})$ | 空间复杂度 | $O(n^{1.5})$ |

Problem 9

| 试题名称 | Chefbook | | |
|---|---|---|---|
| 试题来源 | Codechef JUNE 15 | 试题编号 | CHEFBOOK |
| 题目大意 | | 算法讨论 | |
| 给定m组限制$a_i, b_i, w_i, l_i, r_i$，你要设置n个非负整数$x_i$和n个非负整数$y_i$，满足$l_i \leq x_{a_i} - y_{b_i} + w_i \leq r_i$，你要最小化$\sum_{i=1}^{m} x_{a_i} - y_{b_i} + w_i$，给出方案。<br>数据范围:$n \leq 100, m \leq n^2$ | | 容易发现这是一个线性规划，我们将其对偶转化到费用流。令$x_{i+n} = y_i$，先把限制写成$x_i - x_j \leq K$的形式，然后从i到j连费用为$K$流量无穷的边，再从原点向i连流量为1费用为0的边，从j向汇点连流量为1费用为0的边，这张图的最小费用最大流就是答案。<br>我们还要构造一组方案。可以利用差分约束系统，如果一条边流量不为0，这个限制就成立，这样就可以构图求答案。 | |
| 时间复杂度 | $O(costflow(n, m) + nm^2)$ | 空间复杂度 | $O(m)$ |

Problem 10

<table>
<tr><td>试题名称</td><td colspan="3">Ciel and Earthquake</td></tr>
<tr><td>试题来源</td><td>Codechef MARCH 12</td><td>试题编号</td><td>CIELQUAK</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">一个$n \times m$的四联通网格，每一条边都有p的概率损毁，问点$(1,1)$和点$(n,m)$联通的概率。<br>数据范围:$T \leq 50, n \leq 8, m \leq 10^{18}$</td><td colspan="2">n很小，显然可以使用轮廓线dp，记录轮廓线以及$(1,1)$点的连通性，利用最小表示法状态数S只有3000多，直接dp是$O(nmS)$的，可是m很大，S也过大导致不能矩阵乘法，但是我们发现当m 比较大的时候大致呈指数级增长，就可以设定一个x，用$Ans(n,x-1) \times (\frac{Ans(n,x)}{Ans(n,x-1)})^{m-x+1}$来估计答案，当x取40的时候就可以通过了。</td></tr>
<tr><td>时间复杂度</td><td>$O(nxs)$</td><td>空间复杂度</td><td>$O(nxs)$</td></tr>
</table>

Problem 11

<table>
<tr><td>试题名称</td><td colspan="3">Attack of the Clones</td></tr>
<tr><td>试题来源</td><td>Codechef JUNE 11</td><td>试题编号</td><td>CLONES</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">我们称一个形为f:A→B的函数叫做布尔函数，其中A是所有长度为n的01串，B={0,1}。<br>现在有四个元素是n项布尔函数的集合：<br>1.Z集合是所有满足f(0,0,...,0)=0的函数的集合。<br>2.P集合是所有满足f(1,1,...,1)=1的函数的集合。<br>3.D集合是所有满足!f(x1,x2,...,xn)=f(!x1,!x2,...,!xn)的函数的集合。<br>4.A集合是所有满足如下条件的函数的集合：如果$f(x1,...,x_{i-1},a,x_{i+1},...,xn) = f(x1,...,x_{i-1},b,x_{i+1},...,xn)$，则$f(y1,...,y_{i-1},a,y_{i+1},...,yn) = f(y1,...,y_{i-1},b,y_{i+1},...,yn)$。<br>现在给你一个由Z,P,D,A,v,^,!,(,),\组成的表达式，其中ZPDA表示如上所述的集合，v表示并集，^表示交集，!表示补集，\表示差集。<br>其中!优先级最高而其余三个运算优先级相同，()的优先级高于!。<br>求出满足这个表达式的元素个数。<br>数据范围:表达式长度$S, n \leq 100$。</td><td colspan="2">用一个4位二进制表示一个函数从属于ZPDA的情况，那么就有16种函数，可以手算出每种函数的数量，然后用一个16位二进制的数来表示表达式是否含有这16种集合，对于集合$X$和$Y$，并集是$X\ or\ Y$，交集是$X\ and\ Y$，补集是$X\ xor$ 65535，差集是$X\ and\ (Y\ xor$ 65535)，然后用栈处理表达式就好了。</td></tr>
<tr><td>时间复杂度</td><td>$O(n+S)$</td><td>空间复杂度</td><td>$O(S)$</td></tr>
</table>

Problem 12

<table>
<tr><td>试题名称</td><td colspan="3">Future of draughts</td></tr>
<tr><td>试题来源</td><td>Codechef AUG 15</td><td>试题编号</td><td>CLOWAY</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定T张$n_i$个点$m_i$条边的无向图。Q组询问，每组询问问只考虑编号为$[L_i, R_i]$的图，最开始对每一张图可以选择一个出发点，接下来每一个回合可以选中一些图（至少选中一个），并对每一个选中的图通过一条存在的边移动一个位置，问在$K_i$回合内每一张图都回到出发点的方案数（对$10^9+7$取模）。<br>数据范围:$n, T \leq 50, K_i \leq 10^4, Q \leq 10^5$</td><td colspan="2">先预处理出每张图长度为k的回路个数。显然长度为k的回路个数等于邻接矩阵G的k次方的对角线上元素的和，直接矩阵乘法是会超时的，可以先对这个邻接矩阵求出它的特征多项式，这个可以根据特征多项式的定义$det(xI-G)$，带入n个不同的x，求行列式然后插值来得到特征多项式$f(x)$，就可以得到一个关于回路个数的n阶线性递推式，这样就可以预处理出长度为k的回路个数了。令$S_{ij}$表示第i张图长度为j的回路个数。<br>通过容斥可以发现恰好K回合的答案是$\sum_{i=1}^{K}(-1)^{K-i}\binom{K}{i}\prod_{j=L}^{R}S_{ji}$，可以枚举L到R然后用FFT计算答案，因为模数很奇怪不能NTT，需要取3个模数进行NTT然后CRT合并。</td></tr>
<tr><td>时间复杂度</td><td>$O(Tn^4+TnK+T^2KlogK)$</td><td>空间复杂度</td><td>$O(Tn^2+K)$</td></tr>
</table>

Problem 13

<table>
<tr><td>试题名称</td><td colspan="3">Counting D-sets</td></tr>
<tr><td>试题来源</td><td>Codechef JAN 14</td><td>试题编号</td><td>CNTDSETS</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">在n维空间中，两个点的距离定义为每一维坐标差的绝对值的最大值，一个点集的直径定义为距离最远的两个点的距离，两个点集是相同的当且仅当它们可以通过平移得到。询问n维空间中直径等于D的点集个数。<br>数据范围:$n \leq 10^3, D \leq 10^9$</td><td colspan="2">直径恰好为D的点集个数不好求，可以求出直径小于等于D的点集个数，答案就$Ans_D - Ans_{D-1}$。为了避免相同的点集可以限制每一维坐标都至少有一个点为0，通过简单的容斥发现答案就是$\sum_{i=0}^{n}(-1)^i\binom{n}{i}2^{(D+1)^{n-i}D^i}$，直接计算即可。</td></tr>
<tr><td>时间复杂度</td><td>$O(nlog(10^9+7))$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 14

<table>
<tr><td>试题名称</td><td colspan="3">Counting Hexagons</td></tr>
<tr><td>试题来源</td><td>Codechef SEPT 11</td><td>试题编号</td><td>CNTHEX</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">现在你有NK根木棍，木棍长度为1到N，且每种长度的木棍有K根。你需要选出六根木棍拼出一个面积为正的六边形。你选取的木棍需要满足最长的木棍长度至少为L，其它的木棍长度不能超过X。求方案数<br>数据范围:$1 \leq K \leq 5, 1 \leq X < L \leq N \leq 10^9, N - L \leq 100$</td><td colspan="2">N-L很小，可以枚举最长的木棍长度，其他木棍长度和要超过最长木棍长度，为了避免0的出现可以先把所有木棍长度-1。<br>我们可以使用二进制数位dp，令$f_{a,b,c,d,e}$表示考虑后a位，当前长度和与最长木棍长度关系为b，长度和进位为c，5根木棍大小关系为d，5根木棍与X的关系，接下来枚举每一位是0还是1即可。</td></tr>
<tr><td>时间复杂度</td><td>$O(4K^2(N-L)logNBell_K)$</td><td>空间复杂度</td><td>$O(K(N-L)2^KBell_K)$</td></tr>
</table>

Problem 15

<table>
<tr><td>试题名称</td><td colspan="3">Find a special connected block</td></tr>
<tr><td>试题来源</td><td>Codechef APRIL 12</td><td>试题编号</td><td>CONNECT</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一个$n \times m$的网格，每一个格子都有一个$[-1, n \times m]$范围内的整数权值以及一个代价。问一个代价和最小的四联通块，满足联通块中没有权值为-1的格子且至少出现了k种不同的正权值。<br>数据范围:$n, m \leq 15, k \leq 7$</td><td colspan="2">如果权值在$[-1, k]$之间就可以用经典的斯坦纳树算法解决，那么我们可以随机一个$[-1, n \times m]$到$[-1, k]$的映射求斯坦纳树，显然结果不会更优，而且有$\frac{k!}{k^k}$的概率得到最优解，随机x次就可以了。</td></tr>
<tr><td>时间复杂度</td><td>$O(xnm(3^k + 2^k logn))$</td><td>空间复杂度</td><td>$O(nm2^k)$</td></tr>
</table>

Problem 16

<table>
<tr><td>试题名称</td><td colspan="3">Cool Numbers</td></tr>
<tr><td>试题来源</td><td>Codechef JUNE 12</td><td>试题编号</td><td>COOLNUM</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">一个数A是cool number当且仅当可以选出至多三个不同的数位，令这些数位的和为S，A的数位和为K，存在一种选法使得$A|(K-S)^S$。给一个n求出小于等于n的最大cool number以及大于n的最小cool number。<br>数据范围:$T \leq 10^5, n \leq 10^{1000}$,保证所有数据中n的位数和不超过$4 \times 10^6$。</td><td colspan="2">如果一个数只有不超过三个非0位，显然满足条件，也很容易求得小于等于n的最大这类数以及大于n最小的数。<br>对于剩下的数，令A的位数为k，可以发现$(9k-27)^{27} > 10^{k-1}$，k最大只有77，于是可以枚举K-S，然后枚举$(K-S)^2$7的因子判断，总共只有30000多个，询问时二分就好了。</td></tr>
<tr><td>时间复杂度</td><td>$O(Tlognlog30000)$</td><td>空间复杂度</td><td>$O(logn + 30000)$</td></tr>
</table>

Problem 17

| 试题名称 | Count on a Treap | | |
|---|---|---|---|
| 试题来源 | Codechef FEB 14 | 试题编号 | COT5 |
| 题目大意 | | 算法讨论 | |
| 有一个初始为空的treap，进行n次操作，操作种类如下：<br>1. 插入一个给定关键字和优先级的节点。<br>2. 删除一个节点。<br>3.询问两个节点之间的路径长度。<br>保证关键字和优先级两两不同。<br>数据范围:$n \le 2 \times 10^5$ | | treap的中序遍历就是关键字的顺序，两个点的LCA就是他们对应区间中优先级最大的点，那么现在的问题就是求一个点的深度。<br>如果x是y的祖先，也就是$LCA(x,y) = x$，那么$[x,y]$区间中x的优先级最大，我们可以用线段树来维护某个点左端（右端）的祖先个数，我们考虑一个函数$getl(x,y)$，表示对于线段树中的区间x，求$max$(前驱最大值$,y)$ =当前点权值的点数，只要维护了区间最大值这个函数很容易实现，复杂度为$O(logn)$，同时使用$getr(x,y)$就可以满足查询和更新了。单次复杂度$O(log^2n)$。 | |
| 时间复杂度 | $O(nlog^2n)$ | 空间复杂度 | $O(n)$ |

Problem 18

| 试题名称 | Arithmetic Progressions | | |
|---|---|---|---|
| 试题来源 | Codechef NOV 12 | 试题编号 | COUNTARI |
| 题目大意 | | 算法讨论 | |
| 给定一个长度为n数组A，询问有多少对$(i,j,k)(1 \le i < j < k \le n)$满足$A_i + A_k = 2A_j$。<br>数据范围:$n \le 10^5, A_i \le 3 \times 10^4$ | | 可以使用分块，设块大小为$size$，分成$\frac{n}{size}$块，对于$(i,j,k)$所属块的情况讨论。<br>1.至少有两个在同一块：每块都枚举两个数，再预处理一个前缀桶和后缀桶，就可以求出。这一部分是$O(size^2 \times \frac{n}{size})$的。<br>2.所有都不在同一块：枚举j所在的块，把前缀桶和后缀桶用FFT乘起来，就可以查找了，这一部分是$O(\frac{n}{size}nlogn)$的。<br>所以复杂度是$O(size^2 \times \frac{n}{size} + \frac{n}{size}nlogn)$，令$size = \sqrt{nlogn}$，复杂度最优，为$O(n\sqrt{nlogn})$ | |
| 时间复杂度 | $O(n\sqrt{nlogn})$ | 空间复杂度 | $O(n)$ |

Problem 19

<table>
<tr><td>试题名称</td><td colspan="3">Cucumber Boy and Cucumber Girl</td></tr>
<tr><td>试题来源</td><td>Codechef JAN 13</td><td>试题编号</td><td>CUCUMBER</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定m个$n \times n$的矩阵$A_i$，对于数对$(a,b)(a < b)$定义$n \times n$的矩阵B满足$B_{i,j} = \sum_{k=1}^{n} A_{a,i,k}A_{b,j,k}$，一个1-n的排列P是好的当且仅当至少存在一个i使得$B_{i,p_i}$是奇数，数对$(a,b)$是好的当且仅当好的排列有奇数个。询问有多少个好数对。<br>数据范围:$n \le 60, m \le 8000$</td><td colspan="2">对于矩阵B，令矩阵C为$C_{i,j} = (B_{i,j} + 1)\ mod\ 2$。考虑矩阵C的行列式，每一个不是好的排列的贡献都是1或-1，每一个好的排列的贡献都是0，当$n > 1$的时候数对$(a,b)$是好的等价于$det(C)$为奇数。<br>因为$C_{i,j} = (\sum_{k=1}^{n} A_{a,i,k}A_{b,j,k} + 1)\ mod\ 2$，所以可以在每一个矩阵A后面补上全是1的第n+1列，那么$C = A_a \times A_b^T$（模2意义下）。<br>现在的问题就是求$det(C)$，由Binet-Cauchy 定理，令$A_{i-j}$为矩阵$A_i$删掉第j列后得到的矩阵，那么就有$det(C) = \sum_{i=1}^{n+1} det(A_{a-i}) + det(A_{b-i})$。所以只需要求出所有的$det(A_{a-i})$即可。我们可以对$A_i$进行消元，如果$A_i$不满秩，那么$det(A_{i-j})$一定为0。否则一定可以把$A_i$消成$n \times n$的单位矩阵加上一列的形式（令消元后的矩阵为D），假设加上的是第k列，那么就有$det(A_{i-j}) = \begin{cases} D_{j,k} & j < k \\ 1 & j = k \\ 0 & j > k \end{cases}$ 于是就可以计算答案了，注意使用bitset优化。</td></tr>
<tr><td>时间复杂度</td><td>$O(n^2m + m^2)$</td><td>空间复杂度</td><td>$O(n^2m)$</td></tr>
</table>

Problem 20

<table>
<tr><td>试题名称</td><td colspan="3">Payton numbers</td></tr>
<tr><td>试题来源</td><td>Codechef FEB 15</td><td>试题编号</td><td>CUSTPRIM</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">定义三元组$(a,b,c)$的乘法运算，其中c=11或者24。定义单位元A，对于任何B都满足$A \times B = B$。定义zeroA，对任何B都满足$zeroA \times B = zeroA$。定义一个三元组是素数当且仅当这个三元组不能表示成两个非零非单位元的三元组的乘积。<br>给定一个三元组，判断它是不是素数。<br>数据范围:$T \leq 10^4, -10^7 \leq a, b \leq 10^7$</td><td colspan="2">令$\omega = \frac{1+\sqrt{-11}}{2}$，那么对于每一个三元组$(a,b,c)$，都有到域$Z[\omega]$的映射$\phi(a,b,c) = (33-2a-c)+(b-a)\omega$。于是问题就转化为了判断域$Z[\omega]$下的数$a+b\omega$是否为素数。<br>定义共轭$(a+b\omega)c = (a+b-b\omega)$，那么就有如下结论：<br>1.如果x不是整数，那么x是质数当且仅当$x \times x'$是质数。<br>2.如果x是整数，那么x是质数当且仅当x是质数且要么$|x| = 2$，要么$|x| \neq 11$且-11在模x域下没有二次剩余。<br>于是可以直接用欧拉判别法和miller rabin 来判断。</td></tr>
<tr><td>时间复杂度</td><td>$O(Tloga)$</td><td>空间复杂度</td><td>$O(1)$</td></tr>
</table>

Problem 21

<table>
<tr><td>试题名称</td><td colspan="3">Graph Challenge</td></tr>
<tr><td>试题来源</td><td>Codechef FEB 14</td><td>试题编号</td><td>DAGCH</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一个n个点m条边的图，每一个点的标号为它的DFS序。保证所有节点都能从第一个点到达。一个节点x对y来说是好的当且仅当$x < y$且存在一条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来说是最好的当且仅当它是所有对y的好节点中编号最小的。<br>给定Q个询问，问对于一个节点，它是多少个节点的最好的节点。<br>数据范围:$1 \leq n, Q \leq 10^5, n-1 \leq m \leq 2 \times 10^5$</td><td colspan="2">可以发现一个节点的最好的节点就是dominator tree算法中的半必经点。所以只需要运行dominator tree 算法然后统计答案。</td></tr>
<tr><td>时间复杂度</td><td>$O((n+m)\alpha(n))$</td><td>空间复杂度</td><td>$O(n+m)$</td></tr>
</table>

Problem 22

<table>
<tr><td>试题名称</td><td colspan="3">Devu and Locks</td></tr>
<tr><td>试题来源</td><td>Codechef FEB 15</td><td>试题编号</td><td>DEVLOCK</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">对于所有的$0 \leq m \leq M$，求出满足各位数字之和不超过$m$且是$P$的倍数的$n$位数（可以有前导0）。<br>两类数据范围:<br>$n \leq 10^9, P \leq 50, M \leq 500$<br>$n \leq 10^9, P \leq 16, M \leq 15000$</td><td colspan="2">我们把$10^i\ mod\ P$相同的位数放在一起统计，假设当前统计的是$w$，这样的位数有$k$位。<br>令$f_{i,j}$表示模P为i，数字和为j的个数，根据生成函数$g(x) = (\sum_{i=0}^{9} x^i)^k$，可以知道$f_{i \times w\ mod\ P, i} = [x^i]g(x)$，$g(x)$可以利用FFT和快速幂计算。<br>假设原来的答案$h_{i,j}$表示模P为i，数字和为j的个数，现在要把f和h合并，可以得到$h'_{x,y} = \sum_{i+j\ mod\ P=x, a+b=y} h_{i,a} f_{j,b}$，第一维比较小可以暴力，第二维用FFT合并。</td></tr>
<tr><td>时间复杂度</td><td>$O(MP^3 + MP^2 logM)$</td><td>空间复杂度</td><td>$O(MP)$</td></tr>
</table>

Problem 23

<table>
<tr><td>试题名称</td><td colspan="3">Dynamic GCD</td></tr>
<tr><td>试题来源</td><td>Codechef JULY 12</td><td>试题编号</td><td>DGCD</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一棵n个点的带权树，m次操作：<br>1.询问两点之间路径上所有点的gcd。<br>2.给两点之间路径上所有点加一个值。<br>数据范围:$n, m \leq 5 \times 10^4$</td><td colspan="2">首先利用树链剖分转化为序列问题，可以发现$gcd(a, b, c) = gcd(a, b - a, c - b)$，于是可以将序列差分，维护每个点的权值以及区间差分后的gcd，这样gcd就是单点修改了，利用线段树很好维护。</td></tr>
<tr><td>时间复杂度</td><td>$O(mlog^3n)$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 24

<table>
<tr><td>试题名称</td><td colspan="3">Different Trips</td></tr>
<tr><td>试题来源</td><td>Codechef DEC 12</td><td>试题编号</td><td>DIFTRIP</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一棵n个节点的树，每一个节点的权值定义为它的度数，两条路径被视为相同的当且仅当它们长度相同且经过的点的权值构成的字符串相同。询问这棵树有多少条不同的从孩子走向祖先的路径。<br>数据范围:$n \leq 10^5$</td><td colspan="2">直接使用后缀自动机，和字符串不同的地方在于每次加入节点的时候要判断原有节点的孩子是否已被占用。其余都和字符串的后缀自动机算法相同，最后累加一下$len(x) - len(pre(x))$就行了。要注意的是字符集很大需要用map存储孩子。</td></tr>
<tr><td>时间复杂度</td><td>$O(nlogn)$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 25

<table>
<tr><td>试题名称</td><td colspan="3">Simple Queries</td></tr>
<tr><td>试题来源</td><td>Codechef AUG 15</td><td>试题编号</td><td>DISTNUM</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一个长度为n的数列和m个操作：<br>1.定义S为区间[L,R]中出现过的数字的集合，求$\sum_{1\le i<j<k\le|S|} S_iS_jS_k$。<br>2.插入一个数。<br>3.删除一个数。<br>4.修改一个位置的值。<br>5.询问一个区间内出现过的数字种类数<br>数据范围:$n,m \le 10^5$</td><td colspan="2">记一个点的坐标为$(x,y)$，其中x是该点位置，y是x后第一个和x相同的数的位置，可以先用平衡树处理出每个点的坐标，这样只要统计$x \in [L,R], y \in (R,+\infty)$的数，这种在平面上的问题可以使用树状数组套线段树维护。<br>至于1操作的询问，如果记满足条件的点的和为$A$，平方和为$B$，立方和为$C$，答案就是$\frac{A^3-3AB+2C}{6}$，$ABC$都很好维护。<br>需要注意常数问题。</td></tr>
<tr><td>时间复杂度</td><td>$O(mlog^2n)$</td><td>空间复杂度</td><td>$O(mlog^2n)$</td></tr>
</table>

Problem 26

<table>
<tr><td>试题名称</td><td colspan="3">Divide or die</td></tr>
<tr><td>试题来源</td><td>Codechef DEC 14</td><td>试题编号</td><td>DIVIDEN</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">平面上给定一个n度角，你需要通过尺规作图来把这个角分割成n的大小为1度的角。你的操作有：<br>1.画一条连接A,B直线。<br>2.以A为圆心，B,C之间距离为半径画一个圆。<br>你用到的点只能是最开始给定的三个点或者你绘制的图形之间的交点。<br>数据范围:$0 < n < 360$，你的操作次数不能超过1000。</td><td colspan="2">若n是3的倍数则无解，否则有解。<br>先作一个正五边形得到72度的角，再作正三角形得到60度的角，于是得到了12度的角，平分两次得到3度角，用n度角不断减去3度角，最后得到的如果是2度角就平分，于是就得到了一个1度角，不断用这个角划分就好了。</td></tr>
<tr><td>时间复杂度</td><td>$O(n)$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 27

| 试题名称 | Something About Divisors | | |
|---|---|---|---|
| 试题来源 | Codechef AUG 11 | 试题编号 | DIVISORS |
| 题目大意 | | 算法讨论 | |
| 对于给定的正整数B和X，求满足条件的正整数N的个数：对于N，至少存在一个数D($N < D \leq B$)能整除$N \times X$。<br>数据范围:$T \leq 40, X \leq 60, B \leq 10^{12}$ | | 令正整数$i = \frac{NX}{D}$，那么显然有$i < X$，考虑枚举每一个可能的i，为了避免重复的统计，我们可以计算满足$i \mid NX$且不存在j满足$i < j < X$且$j \mid NX$的N。<br>因为$i \mid NX$，所以有$A_i = \frac{i}{gcd(i,X)} \mid N$，那么就有$N = A_i k$。<br>因为$j \mid A_i X k$，所以有$B_j = \frac{j}{gcd(A_i X,j)} \mid k$，于是可以对$i < j < X$进行容斥，也就是$Ans_i = \sum (-1)^t \frac{Bi}{A_i X \times LCM(B_j)}$，直接计算是指数级别的，但是记K为可能的LCM种数，K不大，大概$10^4$，所以dp就可以了。 | |
| 时间复杂度 | $O(X^2K)$ | 空间复杂度 | $O(XK)$ |

Problem 28

| 试题名称 | Colored Domino Tilings and Cutsontest | | |
|---|---|---|---|
| 试题来源 | Codechef NOV 11 | 试题编号 | DOMNOCUT |
| 题目大意 | | 算法讨论 | |
| 一个$N \times M$的矩形棋盘。一个棋盘覆盖的染色是指：在棋盘上填上小写字母，使得每个格子有且仅有一个相邻格子的字母和他的一样。棋盘的割是指一条竖直或水平的直线将棋盘分成两半，这条直线不能穿过一对有相同字母的相邻格子。现在你需要构造一个在割数尽可能小的情况下，染色数尽可能小的染色。<br>数据范围:$T \leq 3000, N, M \leq 500$ | | NM为奇数的时候显然无解。令$N \leq M$，当N小于5的时候可以手工构造，当$N \geq 5$的时候可以利用$5 \times 6$和$6 \times 8$的两种棋盘进行扩展，这两种棋盘都是三染色且无割的，可以发现不改变染色数和割数的时候这两个棋盘都很容易扩展两行或两列。注意$6 \times 6$也需要手工构造。 | |
| 时间复杂度 | $O(TNM)$ | 空间复杂度 | $O(NM)$ |

Problem 29

| | | | |
|---|---|---|---|
| 试题名称 | Easy exam | | |
| 试题来源 | Codechef JULY 15 | 试题编号 | EASYEX |
| 题目大意 | | 算法讨论 | |
| 一个K面的骰子，投到每一面的概率都是完全相同的，现在投n次，设投完之后数字i出现了$a_i$次，试求$\sum_{i=1}^{L} a_i^F$的期望。<br>数据范围:$n, K \le 10^9, L \le K, LF \le 5 \times 10^5, F \le 1000$ | | 设$x_{i,j}$为第j次投是否投到i，如果投到就是1否则就是0。那么显然有$a_i = \sum_{j=1}^{n} x_{i,j}$，答案要求的就是$\prod_{i=1}^{L}(\sum_{j=1}^{n} x_{i,j})^F$。我们把这个式子展开，我们可以对每一项分开来考虑，如果一项中存在不同的a,b和c使得$x_{a,c}$和$x_{b,c}$的指数都大于0，那么这一项的贡献就是0，否则设这一项出现了t个不同的变量，那么它的期望值就是$\frac{1}{K^t}$<br>考虑计算有i个贡献不为0的不同变量的项的系数。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，那么显然有递推式$w_{i,j} = w_{i-1,j-1} + j \times w_{i-1,j}$。<br>令$f_{n,m}$为$\prod_{i=1}^{n}(\sum_{j=1} x_{i,j})^F$的展开中出现了m个不同变量系数之和的方案数，那么就有$f_{n,m} = f_{n-1,j} w_{F,m-j} \binom{m}{j}$，这是可以FFT+快速幂优化的，最后统计答案。 | |
| 时间复杂度 | $O(LFlogLFlogL)$ | 空间复杂度 | $O(LF)$ |

Problem 30

| | | | |
|---|---|---|---|
| 试题名称 | Equivalent Suffix Tries | | |
| 试题来源 | Codechef JULT 12 | 试题编号 | EST |
| 题目大意 | | 算法讨论 | |
| 给定一个长度为n的字符串，求由小写字母组成，后缀字母树与该字符串同构的串个数。<br>数据范围:$n \le 10^5$ | | 后缀字母树的性质如下：<br>1.叶子个数取决于最长的是其他后缀前缀的后缀长度，设长度为L，叶子个数就有n-L。<br>2.两个后缀的LCA深度取决于LCP。<br>3.和根相连的点数为字母种数。<br>那么L、字母种数、前n-L-1个后缀和第n-L个后缀的LCP是不变的，其中第三点可以确定一部分字符的相对关系。于是枚举前n-L-1个后缀，使得第n-L个后缀是该后缀的前缀，利用hash判重，最后答案要乘上$\frac{26!}{t!}$，t是字母种数。 | |
| 时间复杂度 | $O(nlogn)$ | 空间复杂度 | $O(n)$ |

Problem 31

| | | | |
|---|---|---|---|
| 试题名称 | Evil Book | | |
| 试题来源 | Codechef MARCH 12 | 试题编号 | EVILBOOK |
| 题目大意 | | 算法讨论 | |
| 有n个人，你打败第i个人需要付出$c_i$的代价，打败它后可以获得$d_i$的魔法值，最开始你的魔法值是0。你可以对人使用魔法，对第i个人使用后$c_i$和$d_i$都将除以3，每一次使用要消耗X点魔法值。你要使你的魔法值大于等于666，问你最少付出多少的代价。<br>数据范围:$T \leq 5, n \leq 10, 10 \leq X \leq 666$ | | 利用贪心思想，如果我们打败一个人时他的魔法值大于666，那么肯定是尽量小的，而X的值至少是10，那么小于666的情况只有3种以内（4次就得不偿失了），所以每个人最多都有4种可能。<br>可以发现存在一种最优解使得使用魔法次数单调不减，按这种方法搜索，加一些剪枝就可以通过了。 | |
| 时间复杂度 | $O(T4^n)$ | 空间复杂度 | $O(n)$ |

Problem 32

| | | | |
|---|---|---|---|
| 试题名称 | Fibonacci Numbers on Tree | | |
| 试题来源 | Codechef SEPT 14 | 试题编号 | FIBTREE |
| 题目大意 | | 算法讨论 | |
| 给定一棵n个节点的树，你要支持以下操作：<br>1.给u到v路径上的点权加上斐波那契数列。<br>2.询问以x为根时y的子树和。<br>3.询问x到y路径上所有节点的权值和。<br>4.让整棵树回到第i次操作后的状态。<br>数据范围:$n, Q \times 10^5$ | | 首先使用树链剖分转化为序列上问题，然后用可持久化线段树维护。<br>如何维护区间加斐波那契数列呢，注意到我们只需要知道开头两个数，就能求出整个斐波那契型数列的和，那么标记就是当前区间加上的开头两个数和末尾两个数，可以利用公式$\sum_{i=1}^{n} fib_i = fib_{n+2} - 1$，剩下的就很好维护了。 | |
| 时间复杂度 | $O(Qlog^2n)$ | 空间复杂度 | $O(Qlog^2n)$ |

Problem 33

| | | | |
|---|---|---|---|
| 试题名称 | Find a Subsequence | | |
| 试题来源 | Codechef FEB 12 | 试题编号 | FINDSEQ |
| 题目大意 | | 算法讨论 | |
| 给定一个长度为5的排列p和一个长度n的整数序列A，现在要求一个A的长度为5的子序列B满足在序列B中恰好有$p_i - 1$ 个数比$B_i$小。<br>数据范围:$T \leq 60, n \leq 2000, A_i \in [-10^9, 10^9]$ | | 我们可以枚举第二个和第四个数，然后可以贪心地选取第一个数和第五个数，如果它比第三个数大就去合法方案里最大的，否则取最小的，然后就得到了第三个数的范围，预处理二维前缀和就可以知道是否存在第三个数，如果存在暴力找出。<br>需要预处理1到i中小于j的最大值，大于j的最小值，小于j的个数，i到n中小于j的最大值，大于j的最小值，这些都可以$O(n^2)$求出。 | |
| 时间复杂度 | $O(Tn^2)$ | 空间复杂度 | $O(n^2)$ |

Problem 34

| | | | |
|---|---|---|---|
| 试题名称 | Flight Distance | | |
| 试题来源 | Codechef FEB 12 | 试题编号 | FLYDIST |
| 题目大意 | | 算法讨论 | |
| 给定一个n个点m条边的图，第i条边的边权是$D_i$，可以修改每条边的边权，要求改过之后也是正数，代价就是改动的大小，求出最小的代价，使得改变之后，每一条边都是两个端点的最短路。输出一个最简分数。<br>数据范围:$n \le 10, m \le \frac{n(n-1)}{2}, 1 \le D_i \le 20$ | | 考虑转化为线性规划，对每条边设两个量$x_i$和$y_i$，表示修改之后是$D_i - x_i + y_i$，考虑最短路求法，限制就是$dis_{i,j} \le dis_{i,k} + dis_{k,j}$和$dis_{u,v} = D_i - x_i + y_i$，变量是$dis_{i,j}$和$x_i, y_i$，要求最小化$\sum x_i + y_i$，于是可以用单纯形法解决。<br>可以发现这个线性规划不存在初始可行解，可以添加辅助单纯形，或者进行一些转化。自定义的分数类型较慢，中间过程可以使用double，最后枚举分母就好了。 | |
| 时间复杂度 | $O(simplex(n^2+m, n^3+m))$ | 空间复杂度 | $O(n^5)$ |

Problem 35

| | | | |
|---|---|---|---|
| 试题名称 | Fibonacci Number | | |
| 试题来源 | Codechef OCT 13 | 试题编号 | FN |
| 题目大意 | | 算法讨论 | |
| 给定M,C，询问满足$fib_n \mod M = C$的最小的n，如果无解输出-1。$fib$表示斐波那契数列。<br>数据范围:$T \le 100, C < M \le 2 \times 10^9$,M为质数且M mod 10为完全平方数。 | | 先写出斐波那契数列的通项公式，可以发现2存在逆元，$\sqrt{5}$也存在等价的整数，于是就可以转化为$x^n - x^{-n} = a$，可以发现$x^n = \frac{a \pm \sqrt{a^2+4}}{2}$，求出模意义下的$\sqrt{a^2+4}$，然后就可以用BSGS算法求解。求模意义下的平方根可以使用Cipolla’s algorithm。 | |
| 时间复杂度 | $O(\sqrt{M})$ | 空间复杂度 | $O(\sqrt{M})$ |

Problem 36

| | | | |
|---|---|---|---|
| 试题名称 | Chef and Churu | | |
| 试题来源 | Codechef NOV 14 | 试题编号 | FNCS |
| 题目大意 | | 算法讨论 | |
| 给定一个长度为n的数组和n个函数$f_i = \sum_{j=L_i}^{R_i} A_j$。接下来有m个操作：<br>1.单点修改A的值。<br>2.询问$\sum_{i=L}^{R} f_i$。<br>数据范围:$n, m \le 10^5$ | | 考虑分块，每次询问就是一段整块的和以及$O(\sqrt{n})$次查询A的前缀和，我们需要维护每个块的f值的和，以及$O(1)$查询A的前缀和。<br>维护每个块的f值的和可以先预处理一个数组$Num_{i,j}$表示$A_j$在第i块中的出现次数，预处理是$O(n\sqrt{n})$，修改单点之后枚举每块修改。<br>维护A的前缀和也可以对A分块，维护$S1_i$表示前i块的和，以及$S2_i$表示i到i所属块的开头的和，这样一次修改就是$O(\sqrt{n})$了，询问可以用S1和S2做到$O(1)$。 | |
| 时间复杂度 | $O(n\sqrt{n})$ | 空间复杂度 | $O(n\sqrt{n})$ |

Problem 37

| 试题名称 | Chef and Graph Querie | | |
|---|---|---|---|
| 试题来源 | Codechef MARCH 14 | 试题编号 | GERALD07 |
| 题目大意 | | 算法讨论 | |
| 给定一张n个点m条边的无向图，Q个询问，询问只保留编号在区间$[L_i, R_i]$中的边的话图中有多少个联通块。<br>数据范围:$n, m, Q \le 2 \times 10^5$ | | 编号从小向大依次添加边，用LCT维护编号最大的生成树，那么加入第$r$条边后处理$[l, r]$的询问，答案就是$n - t$，t表示生成树中有多少编号大于等于l的边，这个使用一个树状数组维护就可以了。 | |
| 时间复杂度 | $O((m+Q)logn)$ | 空间复杂度 | $O(n)$ |

Problem 38

| 试题名称 | Chef and Tree Game | | |
|---|---|---|---|
| 试题来源 | Codechef APRIL 14 | 试题编号 | GERALD08 |
| 题目大意 | | 算法讨论 | |
| 有一棵n个节点的树，树上每一条边的颜色都为红色或者蓝色。现在两个人轮流进行操作，第一个人每次选择一条红色边删除，第二个人每次选择一条蓝色边删除，删除后和树根不连通的部分将被删除，若干轮之后不能操作的人算输。<br>如果两个人都使用最优策略，问第一个人先手时、第二个人先手时分别是谁赢得游戏。<br>数据范围:$n \le 10^5$ | | 这是不平等博弈，我们对每个节点定义一个局面函数$f$。<br>叶子的函数值为0。每一个节点的函数值为所有孩子的贡献之和，对于一个孩子i，如果连接它的是红边，那么令a为$f_i + a > 1$的最小正整数，它的贡献就是$\frac{f_i+a}{2^{a-1}}$，否则令a为$f_i - a < -1$的最小正整数，它的贡献就是$\frac{f_i-a}{2^{a-1}}$。<br>如果根节点的函数值是为0，那么谁后手谁赢；如果是整数，那么无论如何都是第一个人赢；否则无论如何都是第二个赢。<br>发现需要维护二进制小数，而一个小数的1个数不会超过子树大小，那么就可以用平衡树维护每个1位，并使用启发式合并来做加法。 | |
| 时间复杂度 | $O(nlog^2n)$ | 空间复杂度 | $O(nlogn)$ |

Problem 39

| | | | |
|---|---|---|---|
| 试题名称 | Game of Numbers | | |
| 试题来源 | Codechef JULY 14 | 试题编号 | GNUM |
| 题目大意 | | 算法讨论 | |
| 给定两个长度为n数组A,B，每次要选出两个数对$(i,j)$和$(p,q)$满足$A_i < B_j, A_p > B_q, gcd(A_i, B_j, A_p, B_q) > 1$，$(i,j)$和$(p,q)$都未被选择过。问最多能选择多少次。<br>数据范围:$n, m \leq 400$ | | 枚举每一个$gcd(A_i, B_j) > 1$的数对，可以根据$A_i$和$B_j$的大小把所有数对分成两类。对于图中两个点，如果满足这两个点的gcd大于1，就在它们之间连一条边，那么这张图的最大流就是答案。但是这样边数是$O(n^4)$的，需要优化。<br>对于质数p，我们新建一个点，并对于所有gcd是p的倍数的点和这个点连一条边，这样和原图是等价的。还可以合并一些相同的点来优化。 | |
| 时间复杂度 | $O(maxflow(n^2, n^2logn))$ | 空间复杂度 | $O(n^2logn)$ |

Problem 40

| | | | |
|---|---|---|---|
| 试题名称 | Counting on a directed graph | | |
| 试题来源 | Codechef MAY 15 | 试题编号 | GRAPHCNT |
| 题目大意 | | 算法讨论 | |
| 给定一张n个点m条边的有向图，一个点对$(X,Y)$是合法的当且仅当存在一条从1到X的路径，一条从1到Y的路径满足它们除了1号点以外没有任何公共点。统计合法的$(X,Y)$个数。<br>数据范围:$n \leq 10^5, m \leq 5 \times 10^5$ | | 可以发现两个点合法当且仅当他们没有除了1以外的相同的必经点，也就是在dominator tree上两个点属于1的不同子树，于是可以求出dominator tree之后统计答案。 | |
| 时间复杂度 | $O((n+m)\alpha(n))$ | 空间复杂度 | $O(n+m)$ |

Problem 41

| | | | |
|---|---|---|---|
| 试题名称 | A Game of Thrones | | |
| 试题来源 | Codechef AUG 12 | 试题编号 | GTHRONES |
| 题目大意 | | 算法讨论 | |
| 有n种数，权值是$u_i$，出现了$c_i$次。两个人轮流操作。第一个回合第一个选取一个数字作为初始值。之后的每一个回合，当前的人需要选出一个和当前值相似的数作为新的值，之前的值移出游戏，如果无法操作则算输。<br>两个数a,b是相似的当且仅当$b\|a$且$\frac{a}{b}$是质数。你需要判断哪个人必胜，如果是第一个人必胜，你需要输出可以使他获胜的最小的初始值是多少。<br>数据范围:$n \leq 500, u_i \leq 10^18, c_i \leq 10^9$ | | 先用miller rabin判断质数，求出任意两个数是否相似。把相似的数连上边，那么第二个人必胜当且仅当存在完备匹配，可以证明这个图是二分图，那么可以使用最大流求出谁必胜。<br>如果一个初始权值可以让第一个人获胜，那么一定存在一个最大匹配不匹配这个权值，每次网络流太慢，可以使用退流。 | |
| 空间复杂度 | $O(nlogn)$ | 时间复杂度 | $O(n^2logn + maxflow(n, nlogn))$ |

Problem 42

| | | | |
|---|---|---|---|
| 试题名称 | A game on a graph | | |
| 试题来源 | Codechef JULY 15 | 试题编号 | HAMILG |
| 题目大意 | | 算法讨论 | |
| 给定一张无向图，两个人玩游戏。第一个选择一个出发点，接着两个玩家轮流操作，每一次操作可以沿着边移动到一个未被到达过的点，无法移动的人会输掉这个游戏。两个人都以最优策略移动，问有多少个出发点对第一个人来说是必胜的。<br>数据范围:$T \leq 3, n \leq 2000, m \leq 3 \times 10^5$ | | 一个点是对于第一个人是必胜的当且仅当存在一个原图的最大匹配使得这个点在这个匹配中是未被匹配。于是我们可以先用带花树求一遍最大匹配。可以发现和当前的任意未被匹配的点之间存在一条长度为偶数的增广路径的所有点都可能在一次增广之后变成未被匹配的点。于是我们可以从每一个未被匹配的点出发再做一次增广，把所有距离为偶数或者花中的节点都标记为必胜。最后统计点数就是答案了。 | |
| 时间复杂度 | $O(Tnm)$ | 空间复杂度 | $O(n+m)$ |

Problem 43

| 试题名称 | Knight Moving | | |
|---|---|---|---|
| 试题来源 | Codechef SEPT 12 | 试题编号 | KNGHTMOV |
| 题目大意 | | 算法讨论 | |
| 一个无限大的棋盘，有一个骑士在$(0,0)$。如果骑士的坐标为$(x,y)$，那么下一个回合可以移动到$(x+A_x,y+A_y)$或$(x+B_x,y+B_y)$。有$K$个障碍物。问有多少种方案把骑士移动到$(X,Y)$，无穷大输出-1。<br>数据范围:$K \le 15$，其余数字的绝对值不超过$d=500$。 | | 若$(A_x,A_y)$和$(B_x,B_y)$线性无关，那么到每一个格子需要的两种方法的步数都是唯一的，比如需要走$a$步$A$和$b$步$B$，如果没有障碍物，那么方案数就是$\binom{a+b}{a}$，有障碍物就容斥一下。<br>若线性相关，那么就是一条直线上的问题，只需要在$[-d,d]$之间dp，走$2d$步就可以了，如果再走一步答案变多的话就是-1。 | |
| 时间复杂度 | $O(K^2+d^2)$ | 空间复杂度 | $O(d^2)$ |

Problem 44

| 试题名称 | Little Elephant and Boxes | | |
|---|---|---|---|
| 试题来源 | Codechef MAY 12 | 试题编号 | LEBOXES |
| 题目大意 | | 算法讨论 | |
| 有n个盒子，第i个盒子里有$P_i$的概率是$V_i$块钱，$1-P_i$的概率是一块钻石。你打开了所有盒子之后去买东西，一共有m件物品，第i件需要$C_i$块钱和$D_i$个钻石，你一定会买数量尽可能多的物品。问你期望能买到多少件物品。<br>数据范围:$n,m,D_i \le 30, V_i, C_i \le 10^7$ | | 先用用背包求出用a块钻石买b个物品最少需要多少钱。然后可以使用meet in middle，先处理出前一半物品和后一半物品的所有情况，枚举前一半有多少钻石和后一半有多少钻石以及买多少物品，然后把前一半的状态和后一半的状态按照钱数排序，用两个指针扫描并统计答案。 | |
| 时间复杂度 | $O(nm^2+nm2^{\frac{n}{2}})$ | 空间复杂度 | $O(2^{\frac{n}{2}}+nm)$ |

Problem 45

| 试题名称 | Little Elephant and Colored Coins | | |
|---|---|---|---|
| 试题来源 | Codechef MARCH 13 | 试题编号 | LECOINS |
| 题目大意 | | 算法讨论 | |
| 给定n种给定面值$V_i$和颜色$C_i$的硬币，每种硬币都有无穷多个。Q组询问，要选出一些硬币使得它们的和为S，需要最大化选出硬币中的颜色种类数。<br>数据范围:$n \le 30, V_i, Q \le 2\times 10^5, S \le 10^18$ | | 我们令m是最小的面值，那么可以在模m的意义下进行背包，设$f_{i,j}$表示前i种颜色，可以拼成模m为j的最小面值，要注意特判是否包括最小面值的硬币的颜色。 | |
| 时间复杂度 | $O(n^2m)$ | 空间复杂度 | $O(nm)$ |

Problem 46

<table>
<tr><td>试题名称</td><td colspan="3">Little Party</td></tr>
<tr><td>试题来源</td><td>Codechef APRIL 15</td><td>试题编号</td><td>LPARTY</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定m个长度为n的01串，现在你需要使用一些基子集，使得所有给定的串都可以被这些基子集覆盖且没有给定的串都没有被这些基子集覆盖。你需要最小化使用的基子集的大小的和。<br>数据范围:$T \leq 120, m \leq 1000, n \leq 5$</td><td colspan="2">这是一个NPC问题，所以只能考虑搜索，先记录下所有可行的基子集，至多有$3^n$个，直接搜索的复杂度$O(2^{3^n})$，需要优化。<br>如果一个基子集被另一个包含，那么这个就没有意义，这样最多只会剩下32个基子集，然后加一点剪枝就可以通过了。</td></tr>
<tr><td>时间复杂度</td><td>$O(T2^{3^n})$</td><td>空间复杂度</td><td>$O(3^n + m)$</td></tr>
</table>

Problem 47

<table>
<tr><td>试题名称</td><td colspan="3">Luckdays</td></tr>
<tr><td>试题来源</td><td>Codechef NOV 11</td><td>试题编号</td><td>LUCKYDAY</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定整数A,B,X,Y,Z,P,C，按照以下的方式生成序列$S_1 = A, S_2 = B, S_i = (XS_{i-1} + YS_{i-2} + Z) \, mod \, P$。Q组询问，每组询问给出$L, R$，你需要求出满足$i \in [L, R]$ 且$S_i = C$的i的个数。<br>数据范围:$T \leq 2, Q \leq 2 \times 10^4, p \leq$ 10007且p是个质数,$L, R \leq 10^{18}$</td><td colspan="2">若X和Y有0，那么循环节小于p，可以暴力处理。否则可以使用矩阵乘法来优化计算递推式，发现这个矩阵存在逆元，那么可以使用BSGS算法计算出循环节。矩阵乘法的结果是$(S_i, S_{i-1}, 1)$，令$S_i = C$，枚举$S_{i-1}$，利用BSGS算出循环节里的位置。最后二分统计答案就可以了。</td></tr>
<tr><td>时间复杂度</td><td>$O(T(p\sqrt{p} + Qlogn))$</td><td>空间复杂度</td><td>$O(p\sqrt{p})$</td></tr>
</table>

Problem 48

<table>
<tr><td>试题名称</td><td colspan="3">Music & Lyrics</td></tr>
<tr><td>试题来源</td><td>Codechef AUG 13</td><td>试题编号</td><td>LYRC</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定n个字符串S和m个字符串T，询问每一个S串在T中一共出现了多少次。<br>数据范围:$n \leq 500, |S| \leq 5000, m \leq 100, |T| \leq 50000$</td><td colspan="2">对S串建AC自动机，然后对于每个T串在AC自动机上运行，把运行到的每个点的权值加1，那么S串的出现次数就是它对应的点的fail树上的子树和。</td></tr>
<tr><td>时间复杂度</td><td>$O(n|S| + m|T|)$</td><td>空间复杂度</td><td>$O(n|S|)$</td></tr>
</table>

Problem 49

<table>
<tr><td>试题名称</td><td colspan="3">Two Magicians</td></tr>
<tr><td>试题来源</td><td>Codechef AUG 12</td><td>试题编号</td><td>MAGIC</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一张n个点m条边的无向图，最开始两个人分别在1号点和2号点。从第一个人开始轮流操作，每一个回合有以下三个步骤：<br>1. 沿着现有的无向边移动任意步，如果这一步结束时两个人在同一个格子，则当前人胜。<br>2.加入一条无向边，如果无法加入则另一个人胜。<br>3. 最开始每一个人有P次传送机会，如果当前人还有传送的机会，他可以选择消耗一次并传送到任意一个节点。<br>问谁必胜。<br>数据范围:$T \le 10^2, n \le 7777, m, p \le 10^5$</td><td colspan="2">考虑朴素的dp思路，需要记录奇数块的个数，偶数块的个数，两人当前块的奇偶性，剩下的传送次数，剩下的不影响连通性的边条数的奇偶性。直接枚举操作进行转移，复杂度是$O(n^2P^2)$，显然不能接受。<br>可以发现每个人至多使用一次传送，当奇数块足够多时dp值以4为周期循环，当偶数块足够多时dp值不变，这里的足够多取10就可以了，dp就可以转化为$O(1)$，先预处理所有的dp状态，每次查表就好。</td></tr>
<tr><td>时间复杂度</td><td>$O(Tm\alpha(n))$</td><td>空间复杂度</td><td>$O(n+m)$</td></tr>
</table>

Problem 50

<table>
<tr><td>试题名称</td><td colspan="3">Martial Arts</td></tr>
<tr><td>试题来源</td><td>Codechef NOV 12</td><td>试题编号</td><td>MARTARTS</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">有一个两边都是n个点的完全二分图，每条边有两个权值$A_{i,j}$和$B_{i,j}$，你要求一个匹配，令A边和为H，B边和为G，对手可以删去你匹配中的一条边，他想最大化G-H，其次最大化G，你想最大化H-G，其次最大化H。<br>数据范围:$n \le 100, A_{i,j}, B_{i,j} \le 10^{12}$</td><td colspan="2">令边权为一个二元组$(A-B, A)$，如果没有对手就可以用KM求出最大匹配。考虑枚举对手会删去哪条边，肯定是最大的那一条。<br>于是把边从大到小排序，先加入所有的边，依次枚举对手删的边。令当前对手要删的边是$e$，先把$e$的权值赋为$+\infty$，表示必选，更新答案，再赋值为$-\infty$，表示以后都不可选。<br>如果每次重新做KM算法复杂度过大，需要维护KM的Label值，每次只需要找一条增广路，这样的单次复杂度是$O(n^2)$的。</td></tr>
<tr><td>时间复杂度</td><td>$O(n^4)$</td><td>空间复杂度</td><td>$O(n^2)$</td></tr>
</table>

Problem 51

| | | | |
|---|---|---|---|
| 试题名称 | Expected Maximum Matching | | |
| 试题来源 | Codechef JUNE 12 | 试题编号 | MATCH |
| 题目大意 | | 算法讨论 | |
| 给定一个$n \times m$个矩阵$p_{i,j}$，生成一个左边n个点右边m个点的二分图，其中左边第i个点和右边第j个点之间有边的概率为$p_{i,j}$，求这样生成的二分图的最大匹配的期望值。<br>数据范围: $n \le 5, m \le 100, p_{i,j} \le 1$ | | n很小，可以使用状态压缩dp，左边总共有$2^n$个子集，我们用一个$2^n$位的二进制数表示状态，表示左边的每个子集是否能和右边匹配，可以发现状态数K不多，不到4000。于是令$f_{i,j}$表示考虑右边的前i个点，当前状态为j的概率，预处理转移直接dp就好了。 | |
| 时间复杂度 | $O(Km)$ | 空间复杂度 | $O(Km)$ |

Problem 52

| | | | |
|---|---|---|---|
| 试题名称 | Max Circumference | | |
| 试题来源 | Codechef OCT 12 | 试题编号 | MAXCIR |
| 题目大意 | | 算法讨论 | |
| 给定二维平面中的三个点ABC和N个操作，第i个操作可以使得A的x坐标增加$x_i$，y坐标增加$y_i$。你可以使用最多K个操作，每一个操作只能使用一次。<br>现在你需要最大化$\|AB\| + \|BC\| + \|AC\|$，答案的绝对误差必须小于$10^{-12}$。<br>数据范围:$K \le N \le 500$ | | 可以发现我们要最大化过点A以BC为焦点的的椭圆，考虑过A的切线，存在两个实数$u, v$使得最大化$uA_x + vA_y$和最大化答案等价。<br>如果已经知道$u, v$，直接按照贡献把操作排序取前K个正贡献操作就可以了。考虑操作顺序发生变化的时刻，只有在$ux_i + vy_i = 0$或者$ux_i + vy_i = ux_j + vy_j$的特殊时刻，那么可以分成$O(n^2)$个部分，每个部分二分一下就可以了。<br>注意到答案要求的精度极高，$sqrt$函数精度不够，我们可以令$\sqrt{S} = I + D$，使用公式$D = \frac{S-I^2}{I+\sqrt{S}}$来得到答案。 | |
| 时间复杂度 | $O(N^2 logN)$ | 空间复杂度 | $O(N^2)$ |

Problem 53

<table>
<tr><td>试题名称</td><td colspan="3">Minesweeper Reversed</td></tr>
<tr><td>试题来源</td><td>Codechef JUNE 11</td><td>试题编号</td><td>MINESREV</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定你一个$R \times C$的扫雷棋盘，其中的雷的位置已经表明。最开始所有的方块都是打开的，你需要关闭所有的方块。你可以通过一次点击来关闭一个方块。在你关闭$(x, y)$后，在正常的扫雷游戏中可能和$(x, y)$同时被打开的格子都会被关闭。现在要你求出至少点击多少次，可以关闭所有的方块。<br>数据范围:$T \le 50, R, C \le 50$</td><td colspan="2">显然每一个雷都要花费一次点击来关闭。我们把剩下的格子分成两类，第一类是和雷相邻的，第二类是不和雷相邻的。其中第二类格子构成了若干个联通块。那么在正常的扫雷中，如果打开了一个联通块，所有和它相邻的第一类格子都会被打开。所以在这儿如果我关闭了一个第一类格子，那么所有和它相邻的联通块以及和这些联通块相邻的第一类格子都会被关闭。<br>可以发现第一类格子至多与两个联通块相邻。把和两个联通块相邻的第一类格子看成边，联通块看成点。那么关闭所有联通块的代价就是联通块数减去这张无向图的匹配数。使用带花树即可。</td></tr>
<tr><td>时间复杂度</td><td>$O(TR^2C^2)$</td><td>空间复杂度</td><td>$O(RC)$</td></tr>
</table>

Problem 54

<table>
<tr><td>试题名称</td><td colspan="3">Misinterpretation 2</td></tr>
<tr><td>试题来源</td><td>Codechef JAN 12</td><td>试题编号</td><td>MISINT2</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">一个长度为n的只包含小写字母的字符串S，如果把它的偶数位依次写到开头，再把奇数位依次写下去，得到的字符串和原串一样，那么称这个字符串是好的。给定L,R，问长度为在L,R之间的好的字符串有多少个。<br>数据范围:$T \le 5, L, R \le 10^{10}, R - L \le 5 \times 10^4$</td><td colspan="2">可以发现重排对应着一个置换，如果这个置换的循环数为$f(n)$，那么长度为n的好的字符串个数就有$26^{f(n)}$个，所以我们可以求出所有的$f(n)$再统计答案。当n为奇数的时候，最后一位不变，而前面的排列和n-1的情况一样，所以此时有$f(n) = f(n-1) + 1$。于是只需要考虑n为偶数的情况。<br>令$ord(p)$为2模p的阶，可以发现所有$gcd(i+1, n) = p$的i每$ord(\frac{n+1}{p})$一组构成了若干个置换，而这样的数有$\phi(\frac{n+1}{p})$个，所以可以得到$f(n) = \sum_{p\|(n+1),p>1} \frac{\phi(p)}{ord(p)}$。<br>首先要对$n+1$质因数分解，我们要分解的连续一段，所以可以预处理$\sqrt{R}$以内的质数直接分解。然后要计算$ord(p)$，发现当$(a, b) = 1$时，存在$ord(ab) = lcm(ord(a), ord(b))$，所以可以预处理质数以及质数的幂，每个n就只要多算一次ord。之后用dfs枚举约数统计答案就好。</td></tr>
<tr><td>时间复杂度</td><td>$O(\frac{R^{\frac{3}{4}}}{logR} + \sqrt{R}logR + T(\frac{\sqrt{R}}{logR} + (R-L)log^2R))$</td><td>空间复杂度</td><td>$O(\sqrt{R} + (R-L)logR)$</td></tr>
</table>

Problem 55

| | | | |
|---|---|---|---|
| 试题名称 | Gangsters of Treeland | | |
| 试题来源 | Codechef NOV 13 | 试题编号 | MONOPLOY |
| 题目大意 | | 算法讨论 | |
| 给定一棵n个节点的有根树，开始每一个点都有一个不同的权值，m次操作：<br>1.把i到根路径上的所有节点标记成一种新的权值。<br>2.询问i子树中所有节点到根路径上不同权值个数的平均值。<br>数据范围:$n, m \leq 2 \times 10^5$ | | 修改操作相当于LCT中的access操作，于是可以用LCT维护这棵树，把相同的权值节点当成偏爱孩子，那么access操作次数是$O(nlogn)$的。<br>我们需要维护每个节点到根的不同权值个数，这可以在每次access切换偏爱孩子的时候维护，需要用到dfs序和树状数组。 | |
| 时间复杂度 | $O(nlog^2n)$ | 空间复杂度 | $O(n)$ |

Problem 56

| | | | |
|---|---|---|---|
| 试题名称 | Annual Parade | | |
| 试题来源 | Codechef SEPT 12 | 试题编号 | PARADE |
| 题目大意 | | 算法讨论 | |
| 给定一张n个点m条边的有向图，每个英雄可以从一个$S$移动到一个$T$，$S$可以等于$T$，但他必须向另一个点移动。<br>游行的总费用分成三部分：<br>1.每个英雄路径的边权和，<br>2.若一个英雄的$S \neq T$，需要花费C。<br>3.若一个点没被经过，需要花费C。<br>K个询问，每次询问一个C，求出游行最小花费。<br>数据范围:$n \leq 250, m \leq 30000, K \leq 100000, C, V \leq 10000$。 | | 先用floyd预处理出每两个点的最短路，就可以只考虑英雄经过的关键点了。<br>考虑使用费用流求解，把每个点拆成两个建立一个二分图，从i到j+n连费用为最短距离距离流量为1的边，可以发现如果流量为t，还需要花费$(n-t)C$。那么可以增广n次，记录下每次的费用，对于每个询问都枚举流量进行统计。 | |
| 时间复杂度 | $O(costflow(n, n^2) + Kn + n^3)$ | 空间复杂度 | $O(n^2)$ |

Problem 57

| | | | |
|---|---|---|---|
| 试题名称 | Sine Partition Function | | |
| 试题来源 | Codechef OCT 11 | 试题编号 | PARSIN |
| 题目大意 | | 算法讨论 | |
| 求$\sum_{\sum_{i=1}^{m} k_i = n} \prod_{i=1}^{m} sin(k_i K)$，其中n,m,K已给定，$k_i$为非负整数。<br>数据范围:$m \leq 50, n \leq 10^9, K \leq 6.28$ | | 考虑DP，令$f_{n,m} = \sum_{\sum_{i=1}^{m} k_i = n} \prod_{i=1}^{m} sin(k_i K)$<br>因为$sin(k_i K) = 2cos(K)sin((k_i-1)K) - sin((k_i-2)K)$<br>因此$f_{n,m} = sin(K)f_{n-1,m-1} + 2cos(K)f_{n-1,m} - f_{n-2,m}$，于是就可以用矩阵优化dp转移。 | |
| 时间复杂度 | $O(m^3logn)$ | 空间复杂度 | $O(m^3)$ |

Problem 58

<table>
<tr><td>试题名称</td><td colspan="3">Prime Distance On Tree</td></tr>
<tr><td>试题来源</td><td>Codechef AUG 13</td><td>试题编号</td><td>PRIMEDST</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一棵$n$个节点的树，在树上随机的选一条路径$(u, v)(u < v)$，问路径长度是质数的概率是多少。<br>数据范围:$n \leq 50000$</td><td colspan="2">考虑求出长度为$i$的路径有多少条。可以点分治，每一次合并两个部分的时候发现合并方式类似卷积，于是可以用FFT优化，为了保证复杂度可以按照子树大小从小往大依次合并，也可以把所有子树的和平方再减去每棵子树的平方。</td></tr>
<tr><td>时间复杂度</td><td>$O(nlog^2n)$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 59

<table>
<tr><td>试题名称</td><td colspan="3">Push the Flow!</td></tr>
<tr><td>试题来源</td><td>Codechef AUG 14</td><td>试题编号</td><td>PUSHFLOW</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一个n个点m条边的简单无向连通图，每个点只属于一个简单环，q个操作：<br>1.修改某条边的边权。<br>2.询问以S为源点，T为汇点的最大流。<br>数据范围:$n \leq 10^5, m, q \leq 2 \times 10^5$</td><td colspan="2">最大流等于最小割，所以我们就是要求最小的几条边使得S和T不连通。<br>如果把环缩成点，那么原图就是一颗树，割只可能是S到T路径上的某个环上的两条边或者连接两个环的树边，树边用树链剖分很好维护。树链剖分的同时可以处理出每个点重儿子经过这个环到的两条边，那么只有log次不知道，再利用一个线段树维护就好了。</td></tr>
<tr><td>时间复杂度</td><td>$O(nlog^2n)$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 60

<table>
<tr><td>试题名称</td><td colspan="3">Queries With Points</td></tr>
<tr><td>试题来源</td><td>Codechef NOV 13</td><td>试题编号</td><td>QPOINT</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">平面上给出n个简单$K_i$边形。接着在线的给出Q次询问，给出一个点，你需要求出这个点在哪个多边形内。<br>数据范围:$n, Q \leq 10^5, \sum K_i \leq 3 \times 10^5$</td><td colspan="2">若没有强制在线，可以使用经典的扫描线算法来解决点定位问题，按x坐标从小到大扫描，用平衡树维护线段，在两个关键点直接线段的相对顺序不会变。查询的时候找到该点上面的线段和下面的线段就可以了。<br>因为有强制在线，需要用可持久化平衡树来记录下所有的平衡树，每次按x坐标二分出这个点属于哪一个平衡树，然后查询。</td></tr>
<tr><td>时间复杂度</td><td>$O((K + Q)logK)$</td><td>空间复杂度</td><td>$O(KlogK)$</td></tr>
</table>

Problem 61

| 试题名称 | Rectangle Query | | |
|---|---|---|---|
| 试题来源 | Codechef SEPT 14 | 试题编号 | QRECT |
| 题目大意 | | 算法讨论 | |
| 在一个二维平面中，你需要支持以下三种操作：<br>1. 插入一个矩形。<br>2. 删除一个矩形。<br>3. 给定一个矩形，询问当前有多少个矩形与这个矩形相交。<br>数据范围:$Q \le 10^5$ | | 先对整个操作序列CDQ分治，问题就变成了插入一些矩形，接着询问有多少个给定矩形与给定矩形有交点。<br>可以把一个矩形拆成两条竖直线段，对x坐标使用扫描线，统计和区间[l,r]相交的区间[a,b]数目可以表达为区间总数-$b < l$的数目-$a > r$的数目，用树状数组就可以维护。 | |
| 时间复杂度 | $O(Qlog^2Q)$ | 空间复杂度 | $O(Q)$ |

Problem 62

| 试题名称 | Queries on tree again! | | |
|---|---|---|---|
| 试题来源 | Codechef MAY 13 | 试题编号 | QTREE |
| 题目大意 | | 算法讨论 | |
| 给定一棵n个节点的基环树，保证环的大小是奇数。m次操作：<br>1.把两点路径上的边权取相反数。<br>2.询问两点路径上的最大子段和。<br>数据范围:$n, m \le 10^5$ | | 如果是普通的树，直接使用树链剖分就可以处理。那么我们任意断开环上的一条边，然后每次的操作都判断是否经过这条边（按照长度判断），操作可以变成一次树上路径操作或者两次树上路径操作+一次特殊边操作。注意信息是有序的。 | |
| 时间复杂度 | $O(nlog^2n)$ | 空间复杂度 | $O(n)$ |

Problem 63

| 试题名称 | Query on a tree VI | | |
|---|---|---|---|
| 试题来源 | Codechef DEC 13 | 试题编号 | QTREE6 |
| 题目大意 | | 算法讨论 | |
| 给定一棵$n$个点的树，点有黑色或白色，m个操作：<br>1.修改一个点的颜色。<br>2.询问一个点所在同色联通块的大小。<br>数据范围:$n, m \le 10^5$ | | 维护每个点假设它是黑（白）色，只考虑它子树的黑（白）色联通块大小，那么只要对询问点找到联通的深度最浅的祖先就可以了。我们可以用LCT维护，修改其中的Access操作，让LCT里的每个splay里的点都同色。那么只要Access(x)就可以找到联通的深度最浅的祖先。修改的时候发现splay需要维护区间修改，打个标记就可以了。 | |
| 时间复杂度 | $O(nlogn)$ | 空间复杂度 | $O(n)$ |

Problem 64

| | | | |
|---|---|---|---|
| 试题名称 | Observing the Tree | | |
| 试题来源 | Codechef FEB 13 | 试题编号 | QUERY |
| 题目大意 | | 算法讨论 | |
| 给定一棵n个节点的树，要支持以下三种操作：<br>1.路径加等差数列。<br>2.询问路径和。<br>3.恢复到第i 次修改后的情况。<br>数据范围:$n, m \le 10^5$ | | 首先进行树链剖分，转化为序列上问题。然后用可持久化线段树来维护，对线段树上每一个节点记录一个$k, b$表示对这个区间加上首项为b公差为k的等差数列，这个标记是可以合并的，也可以进行标记永久化，所以就能维护了。 | |
| 时间复杂度 | $O(nlog^2n)$ | 空间复杂度 | $O(nlog^2n)$ |

Problem 65

| | | | |
|---|---|---|---|
| 试题名称 | Ranka | | |
| 试题来源 | Codechef JAN 15 | 试题编号 | RANKA |
| 题目大意 | | 算法讨论 | |
| 你需要在一个$9 \le 9$的围棋棋盘上，构造出一个n步且无重复局面的围棋下法。<br>数据范围:$n \le 10^4$ | | 考虑一种很简单的思路：黑棋先下在除了$(1,1)$的所有地方，然后白棋走$(1,1)$吃掉黑棋，再下在除了$(1,2)$的所有地方，黑棋吃掉白棋，如此循环。这样大概可以走$2*81^2$步，已经足够了。 | |
| 时间复杂度 | $O(n)$ | 空间复杂度 | $O(1)$ |

Problem 66

| | | | |
|---|---|---|---|
| 试题名称 | Petya and Sequence | | |
| 试题来源 | Codechef DEC 13 | 试题编号 | REALSET |
| 题目大意 | | 算法讨论 | |
| 给定一个长度为n的整数序列A，问是否存在一个不全为0的整数序列B满足对于所有的$0 \le j < n$都有$\sum_{i=0}^{n-1} A_i \times B_{i+j \, mod \, n} = 0$。<br>数据范围:$T \le 100, n \le 3 \times 10^4$ | | 问题等价于判断矩阵$X(X_{i,j} = A_{i+j \, mod \, n})$是否满秩。因为X是循环矩阵，令$f(x) = \sum_{i=0}^{n-1} A_i x^i$，若$gcd(f(x), x^n - 1)$ 的次数为d，那么秩就是n-d。定义$\phi_n(x) = \frac{x^n-1}{\prod_{d\|n} \phi_d(x)}$，那么可以发现$\phi_n(x)$两两不等且不可约。所以我们只需要判断是否存在一个n的约数d使得$\phi_d(x)\|f(x)$。这个问题等价于$x^d - 1\|f(x)\prod_{p\|d, p \in prime}(x^{\frac{d}{p}} - 1)$。直接枚举d按照这个式子计算就可以了，可以发现因子数s最多为96。 | |
| 时间复杂度 | $O(ns)$ | 空间复杂度 | $O(n)$ |

Problem 67

| | | | |
|---|---|---|---|
| 试题名称 | Course Selection | | |
| 试题来源 | Codechef DEC 14 | 试题编号 | RIN |
| 题目大意 | | 算法讨论 | |
| 有n门课和m个学期，每一门课都必须选一个学期学习，第i门课在第j个学期学习可以获得$x_{i,j}$的分数。<br>存在K个先后关系，第$A_i$门课必须在第$B_i$门课之前学习。<br>问学完这n门课的分数之和最大是多少。<br>数据范围:$n, m, K \leq 100$ | | 可以利用最小割来解决问题。<br>把第i门课拆成$P_{i,0}, P_{i,1}, ..., P_{i,m}$，总共m+1个点。<br>在$P_{i,j}$和$P_{i,j+1}$直接连$max\{x_{i,j}\} - x_{i,j}$的边。<br>对于限制$a, b$，只要把$P_{a,j}$和$P_{b,j+1}$连$+\infty$的边就可以了。<br>最后用$\sum_i max\{x_{i,j}\}$减去最小割就可以了。 | |
| 空间复杂度 | $O(nm + Km)$ | 时间复杂度 | $O(maxflow(nm, nm + Km))$ |

Problem 68

| | | | |
|---|---|---|---|
| 试题名称 | Random Number Generator | | |
| 试题来源 | Codechef MARCH 15 | 试题编号 | RNG |
| 题目大意 | | 算法讨论 | |
| 有一个k阶线性递推式$A_i = \sum_{j=1}^{K} C_j A_{i-j} \, mod \, 104857601$。给定$A_i, C_i (1 \leq i \leq k)$，求$A_n$。<br>数据范围:$k \leq 3 \times 10^4, n \leq 10^{18}$ | | 线性递推式可以用矩阵乘法优化，但是k过大。可以发现这个矩阵的特征多项式为$f(x) = x^k - \sum_{i=1}^{k} C_i x^{k-i}$，令$g(x) = x^n \, mod \, f(x)$，那么答案就是$\sum_{i=1}^{k} A_i [i-1] g(x)$。求$g(x)$可以用快速幂+多项式取模。 | |
| 时间复杂度 | $O(klognlogk)$ | 空间复杂度 | $O(k)$ |

Problem 69

| | | | |
|---|---|---|---|
| 试题名称 | Room Corner | | |
| 试题来源 | Codechef FEB 13 | 试题编号 | ROC |
| 题目大意 | | 算法讨论 | |
| 给定一个$n \times m$的网格图，现在这个房间所有90度的内角中都站了一个小朋友，小朋友移动的时候必须沿着墙移动。<br>接着游戏开始，每时每刻两个相邻的的小朋友可以交换位置，它们都向对方的位置移动，直到双方同时到达对方的位置才终止。小朋友移动到相邻的空白格需要一个单位的时间。<br>T组询问，询问两个小朋友相遇至少需要多少时间。<br>数据范围:$T \leq 10^4, n, m \leq 2500$ | | 先从一个小朋友开始，逆时针遍历这个环，得到每个小朋友到起始点的距离。这一部分细节很多。<br>两个小朋友肯定是相向不停地交换，那么就有两种可能，对于每一种可能都可以二分一个中间点，计算出时间取最小值就可以了。 | |
| 时间复杂度 | $O(nm + Tlogn)$ | 空间复杂度 | $O(nm)$ |

Problem 70

<table>
<tr><td>试题名称</td><td colspan="3">Sereja and Arcs</td></tr>
<tr><td>试题来源</td><td>Codechef JUNE 14</td><td>试题编号</td><td>SEAARC</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一个长度为n的序列A，询问有多少对$(i,j,k,g)(i<j<k<g),A_i=A_k,A_j=A_g,A_i\neq A_j$。<br>数据范围:$n,A_i\leq 10^5$</td><td colspan="2">因为数对的关系只可能为ABAB,AABB,ABBA三种情况，可以考虑用总数减去AABB和ABBA的数目来得到答案。<br>AABB很容易，只需要预处理前缀和就可以了。<br>对于ABBA，令$size=\sqrt{\frac{n}{log\,n}}$，分两种情况：<br>1. 两种颜色中有一种的数的个数大于$size$，那么就可以通过求出另一种颜色的每一个位置的左侧（右侧）有多少个这种颜色的数，然后把每一对的贡献拆开来累加得到答案。<br>2. 两部分的个数都不多于S，可以发现所有满足这个条件的颜色产生的弧的个数是$O(nsize)$的，所以可以直接使用二维数点的方法来统计这一部分的贡献。</td></tr>
<tr><td>时间复杂度</td><td>$O(n\sqrt{nlog\,n})$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Problem 71

<table>
<tr><td>试题名称</td><td colspan="3">Sereja and Equality</td></tr>
<tr><td>试题来源</td><td>Codechef JULY 14</td><td>试题编号</td><td>SEAEQ</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">两个数组相似当且仅当元素相对大小相同。<br>两个数组A和B的函数$F(A,B)$等于满足$A[l...r]$和$B[l...r]$相似且逆序对数不超过$E$的$(l,r)$数量。<br>多次询问n,E，求出$\sum F(A,B)$，其中A和B都是一个1到n的排列。<br>数据范围:$n\leq 500,E\leq 10^6,t\leq 10^4$</td><td colspan="2">考虑枚举相似区间长度x，那么位置有$n-x+1$个，如果已经确定了这x个数的顺序，那么一个排列剩下的情况就是$\frac{n!}{x!}$，设$f_{i,j}$表示1 i的排列里逆序对数不超过j的数目，答案就是$(\frac{n!}{x!})^2(n-x+1)f_{x,E}$。<br>考虑如何求$f_{i,j}$，显然有$f_{i,j}=f_{i,j-1}+f_{i-1,j}-f_{i-1,j-i}$，于是先用$O(n^3)$的递推预处理f，每次枚举x统计答案即可。</td></tr>
<tr><td>时间复杂度</td><td>$O(n^3+tn)$</td><td>空间复杂度</td><td>$O(n^3)$</td></tr>
</table>

Problem 72

| 试题名称 | Sereja and Order | | |
|---|---|---|---|
| 试题来源 | Codechef NOV 14 | 试题编号 | SEAORD |
| 题目大意 | | 算法讨论 | |
| 有n个程序和两台电脑，第i个程序在第一台电脑上要运行$A_i$秒，第二台$B_i$秒，现在要在最少的时间内完成所有程序在两台电脑上的运行任务，求出最少时间并输出一个方案。<br>数据范围:$n \le 10^5$ | | 答案的下界是$max(\sum A_i, \sum B_i, A_i + B_i)$，这个下界是可以取到。如果下界是取在$A_i + B_i$，那么可以直接填。否则不好计算，如果已经知道操作顺序求最少时间是可以贪心求解的，可以发现答案方案很多，不断地随机顺序并贪心计算就可以找到解。 | |
| 时间复杂度 | $O(n * K)$，K是随机次数 | 空间复杂度 | $O(n)$ |

Problem 73

| 试题名称 | Sereja and Subsegment Increasings | | |
|---|---|---|---|
| 试题来源 | Codechef MAY 14 | 试题编号 | SEINC |
| 题目大意 | | 算法讨论 | |
| 给定两个长度为n元素在$[0, 3]$的序列A,B，每一次可以选出一个区间$[l, r]$，把$A[l...r]$在模4的意义下加一，问最少进行多少次操作后$A = B$。<br>数据范围:$n \le 10^5$ | | 先把A减去B，然后作差分得到C，现在问题就转化成每次让选择$l, r(l < r)$，令$C[l] + +, C[r] - -$，要求C变成0。<br>这可以使用贪心算法，把C从前向后扫描，记录当前$-1, -2, -3$的数量，如果当前是正数的话就讨论。<br>如果是1，直接让答案+1。<br>如果是2并且有-3，就让这个-3和2配对，答案+1，并得到一个-2，否则答案+2。<br>如果是3并且有-3，就和-3，答案+1，并得到一个-1，否则有-2的话，让答案+2并得到一个-1，否则答案+3。 | |
| 时间复杂度 | $O(n)$ | 空间复杂度 | $O(n)$ |

Problem 74

| 试题名称 | Short | | |
|---|---|---|---|
| 试题来源 | Codechef SEPT 11 | 试题编号 | SHORT |
| 题目大意 | | 算法讨论 | |
| 给定n,k，求满足$n < a,b < k$的$(a,b)$的数量，使得$(a-n)(b-n)\|(ab-n)$。<br>数据范围:$n \le 10^5, n < k \le 10^{18}$ | | 令$a \le b$，那么$b = n + \frac{n(a-1)}{p(a-n)-a}$，那么可以枚举$a$，然后枚举$n(a-1)$的因子$d = p(a-n)-a$，求出$p$和$b$判断是否合法。<br>可以发现$d+a \ge 2(a-n)$，所以$b \le n + \frac{n(a-1)}{a-2n}$，因为$a \le b$，那么$a \le 2n+\sqrt{2n^2-n}$，这样只有$O(n)$个$a$需要枚举。但是这样枚举速度不够，考虑换一种枚举方式。<br>当$a$较大的时候，因为$a \le b$所以$p \le \frac{a^2-n}{(a-n)^2}$，直接枚举p就好了。<br>在$a = n+3000$的时候切换，这时候枚举的次数是$K = 1177$ | |
| 时间复杂度 | $O(nK)$ | 空间复杂度 | $O(nlog\,n)$ |

Problem 75

| 试题名称 | Short II | | |
|---|---|---|---|
| 试题来源 | Codechef DEC 11 | 试题编号 | SHORT2 |
| 题目大意 | | 算法讨论 | |
| 给定质数$p$，问有多少对$a,b(a > p, b > p)$满足$(a-p)(b-p)\|ab$。<br>数据范围:$T \le 5, p \le 10^{12}$ | | 原问题等价于求满足$ab\|p(a+b+p)$的个数，可以按$ab$和$p$的关系分三种情况讨论：<br>1.都被$p$整除，可以发现这样的情况只有$(p\ ,p)$ ,$(p\ ,2p)$ ,$(2p,p),(2p,3p),(3p,2p)$五种。<br>2.都不被$p$整除，令$a < b$，所以$a < 1+\sqrt{p+1}$，$b = \frac{a+p}{ak-1}$，令$d = ak-1$，那么$d\|(a+p), a\|(d+1)$。我们分$b \le d$和$d < b$ 两种情况讨论，枚举较小的数然后判断，枚举的上界是$\sqrt{p+1+\sqrt{p+1}}$。<br>3.有一个被$p$整除，可以发现一个满足第2种情况的解$(a,b)$可以推到$(a, \frac{p(a+p)}{b})$和$(b, \frac{p(b+p)}{a})$两组解，答案就是第2种情况的两倍。 | |
| 时间复杂度 | $O(T\sqrt(p))$ | 空间复杂度 | $O(1)$ |

Problem 76

| 试题名称 | Shortest Circuit Evaluation | | |
|---|---|---|---|
| 试题来源 | Codechef AUG 11 | 试题编号 | SHORTCIR |
| 题目大意 | | 算法讨论 | |
| 给定一个布尔表达式$S$，如果某次已经得到正确的结果，那么就不再继续计算这个表达式。<br>现在给出每一个变量为1的概率，你可以在调整表达式的顺序，使得期望的计算次数最少，求期望概率。<br>数据范围:$\|S\| \leq 30000$。 | | 先建立表达式树，那么我们可以调整每个点的子树顺序，而这些子树都由同一种表达式连接，<br>我们可以进行树形dp，$w_i$和$f_i$分别表示i子树期望运算次数和值为1的概率。如果该点表示and，就将子树按照$\frac{w_i}{1-f_i}$顺序排列最优，如果该点表示and，就将子树按照$\frac{w_i}{f_i}$顺序排列最优。 | |
| 时间复杂度 | $O(\|S\| log \|S\|)$ | 空间复杂度 | $O(\mathrm{S})$ |

Problem 77

| 试题名称 | Team Sigma and Fibonacci | | |
|---|---|---|---|
| 试题来源 | Codechef AUG 14 | 试题编号 | SIGFIB |
| 题目大意 | | 算法讨论 | |
| 求$\sum_{x+y+z=n} 6xyzfib_x fib_y fib_z \, mod \, m$，<br>其中$n, m$为给定值。<br>数据范围:$n \leq 10^{18}, m \leq 10^5$ | | 可以推出答案为$\frac{(5n^5+35n^3-36n^2-4n)fib_n-72n^2fib_{n-1}}{500}$，推导方式很复杂，其中部分方法可以参考http://www.mathstat.dal.ca/FQ/Scanned/15-2/hoggatt1.pdf 的内容。然后将$m$乘上500计算，最后除去500就可以了。 | |
| 时间复杂度 | $O(log\, n)$ | 空间复杂度 | $O(1)$ |

Problem 78

| 试题名称 | Count Special Matrices | | |
|---|---|---|---|
| 试题来源 | Codechef JUNE 13 | 试题编号 | SPMATRIX |
| 题目大意 | | 算法讨论 | |
| 一个$n \times n$的矩阵是好的当且仅当：<br>1.$A_{i,i} = 0$。<br>2.$1 \leq A_{i,j} < n-1 (i \neq j)$"<br>3. 对于每一个整数$0 \leq k \leq n-2$，矩阵中都存在k。<br>4.$A_{i,j} \leq max(A_{i,k}, A_{k,j})$<br>给定n，求好矩阵的个数。<br>数据范围:$n \leq 10^7, T \leq 10^5$ | | 可以得到答案为$\frac{n!(n-1)!}{3\times 2^{n-1}}(\frac{3n}{2} - 2 - \sum_{i=1}^{n-1} \frac{1}{i})$（详细证明可见CC官方题解），于是可以通过$O(n)$预处理来做到$O(1)$ 询问。 | |
| 时间复杂度 | $O(n+T)$ | 空间复杂度 | $O(n)$ |

Problem 79

<table>
<tr><td>试题名称</td><td colspan="3">The Street</td></tr>
<tr><td>试题来源</td><td>Codechef MARCH 14</td><td>试题编号</td><td>STREETTA</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一个长度为$n$的数组$A$和$B$，有$m$次操作<br>1.对于$A$区间加一个等差数列。<br>2.对于$B$区间对一个等差数列取max。<br>3.询问$A_i + B_i$的值。<br>数据范围:$n \le 10^9, m \le 3 \times 10^5$</td><td colspan="2">A数组很好维护，用线段树维护一个区间的首项和公差就好了。<br>维护B数组的时候，可以对一个区间记录一个等差数列，表示对这个等差数列取max，如果有两个标记需要合并，那么考虑这两条直线在这个区间中必然有交点（否则直接保留大的哪一个），那么肯定有一条直线的较大区间只存在于这个区间的一个子树中，递归下去就好了，一次合并标记的复杂度是$O(logn)$。</td></tr>
<tr><td>时间复杂度</td><td>$O(mlog^2n)$</td><td>空间复杂度</td><td>$O(mlogn)$</td></tr>
</table>

Problem 80

<table>
<tr><td>试题名称</td><td colspan="3">String Query</td></tr>
<tr><td>试题来源</td><td>Codechef APRIL 13</td><td>试题编号</td><td>STRQUERY</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定字符串S，n个操作：<br>1.在首或尾或正中间插入一个字符。<br>2.在首或尾或正中间删除一个字符。<br>3.给定一个字符串T，问T在S中出现了多少次。<br>数据范围:$n \le 1.5 \times 10^5, \sum |T| \le 1.5 \times 10^6$</td><td colspan="2">如果只有在一端操作，就可以用后缀平衡树来解决这个问题。<br>如果在首尾操作，就用两个后缀平衡树$L$和$R$来维护，$L$维护前一部分，可以支持首操作，$R$维护后一部分，支持尾操作，如果一个删完的话就暴力重构整个结构。查询的时候中间的部分KMP就好。我们令这个结构为$T$。<br>原题可以用两个$T$结构维护，每一次操作后都进行调整，使得大小平衡，就可以支持中间操作。询问时中间的部分仍然可以用KMP解决。</td></tr>
<tr><td>时间复杂度</td><td>$O((n + \sum |T|) log\, n)$</td><td>空间复杂度</td><td>$O(n + |T|)$</td></tr>
</table>

Problem 81

<table>
<tr><td>试题名称</td><td colspan="3">Counting The Important Pairs</td></tr>
<tr><td>试题来源</td><td>Codechef JAN 14</td><td>试题编号</td><td>TAPAIR</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给定一张n个点m条边的无向图，询问有多少种删掉两条边使得这张图不连通的方案。<br>数据范围:$n \le 10^5, m \le 3 \times 10^5$</td><td colspan="2">选一个生成树，每个非树边赋一个随机值，一条树边的权值是覆盖它的所有非树边的权值的异或。删去一个边集之后不连通的充要条件是存在一个子集的权值异或和为0，那么把所有的边权值排个序统计就可以了。</td></tr>
<tr><td>时间复杂度</td><td>$O(mlog\, m)$</td><td>空间复杂度</td><td>$O(m)$</td></tr>
</table>

Problem 82

| 试题名称 | Selling Tickets | | |
|---|---|---|---|
| 试题来源 | Codechef MAY 12 | 试题编号 | TICKETS |
| 题目大意 | | 算法讨论 | |
| 有$n$道菜$m$个人，每一道菜都给一个人，第$i$个人只要分到$a_i$或$b_i$就开心，否则他就不开心。现在求最大的$x$使得任意的$x$人都存在一个方案使所有人都开心。<br>数据范围:$n \le 200, m \le 500$ | | 把菜看成点，把人看成连接$a_i$和$b_i$的边，那么如果找到一个最小的生成子图$G'(V,E)$满足$\|E\| = \|V\| + 1$，那么答案就是$\|V\| - 1$。这样的子图只有两种情况，于是可以分情况讨论：<br>1.两个点直接有三条路径。直接枚举这两个点并bfs出三条路径就可以了。<br>2.两个环和连接这两个环一条路径。可以枚举一个点$u$使得$u$在路径上，求出bfs树，每个环用一条非树边表示，环的权值是环长+LCA到$u$的距离，取最小值和次小值更新答案。 | |
| 时间复杂度 | $O(n^2m)$ | 空间复杂度 | $O(n+m)$ |

Problem 83

| 试题名称 | Two k-Convex Polygons | | |
|---|---|---|---|
| 试题来源 | Codechef JUNE 13 | 试题编号 | TKCONVEX |
| 题目大意 | | 算法讨论 | |
| 给定$n$根木棍，问能否从中选出$2k$根使得它们可以拼成两个边数为$k$的凸多边形。<br>数据范围:$n \le 1000, k \le 10$ | | 一组木棍可以拼成凸多边形的条件是最长边小于其他边之和，按边长排序，选取木棍最好的情况就是连续的$k$个。<br>因为有长度限制，当$n \ge 70$的时候一定有解，于是就可以先找连续的$k$个删掉，然后再找$k$个。<br>如果第一种方法不成立，最优解一定是连续的$2k$个，那么枚举这$2k$个并暴力枚举集合划分就可以了。 | |
| 时间复杂度 | $O(n^2 + 70k\binom{2k}{k})$ | 空间复杂度 | $O(n)$ |

Problem 84

| 试题名称 | Counting on a Tree | | |
|---|---|---|---|
| 试题来源 | Codechef MARCH 15 | 试题编号 | TREECNT2 |
| 题目大意 | | 算法讨论 | |
| 给定一颗$n$个点带边权的树，$Q$次修改边的权值。每次操作后输出这棵树中所有权值为1的路径条数，路径的权值定义为路径上所有边权值的最大公约数。<br>数据范围:$n \le 10^5, Q \le 100$，边权不超过$Z = 10^6$ | | 离线处理，令$f(i)$为权值为$i$的倍数的条数，那么答案就是$\sum_{i=1}^{Z} f(i)\mu(i)$。<br>假设现在在求$f(i)$，那么我们只需要考虑所有边权为$i$的倍数的边，可以用并查集来求出答案。有修改只需要先把和无修改的边插到并查集中，然后对于每一次修改特殊考虑需要修改的边，然后再复原就好了。因为存在复原操作，所以并查集要用启发式合并。$\mu(i) \neq 0$的$i$个数$K$并不多。 | |
| 时间复杂度 | $O(K(n + Q^2)logn + Z)$ | 空间复杂度 | $O(n)$ |

Problem 85

| 试题名称 | Children Trips | | |
|---|---|---|---|
| 试题来源 | Codechef OCT 14 | 试题编号 | TRIPS |
| 题目大意 | | 算法讨论 | |
| 给定一棵$n$个点的树，边权值都是1或2。Q组询问，询问从$u_i$走到$v_i$，每天最多可以走$d_i$的长度，且每一天的终止点必须是树上的点，需要走几天。<br>数据范围:$n, Q \le 10^5$ | | 可以根据$d$的大小把询问分成两类，$d_i \ge \sqrt{n}$和$d_i < \sqrt{n}$。<br>1.$d_i \ge \sqrt{n}$。答案是$O(\sqrt{n})$的，所以直接用倍增数组模拟每天走的结果就可以了。<br>2.$d_i < \sqrt{n}$。我们把$d_i$相同的询问一起处理，预处理出向上走$2^i$天的结果，然后就可以$O(logn)$处理每一次询问了。 | |
| 时间复杂度 | $O(n\sqrt{n}logn)$ | 空间复杂度 | $O(nlogn)$ |

Problem 86

| 试题名称 | Substrings on a Tree | | |
|---|---|---|---|
| 试题来源 | Codechef APRIL 12 | 试题编号 | TSUBSTR |
| 题目大意 | | 算法讨论 | |
| 给定一棵$n$个节点的树，每个点都有一个字母。一个字符串被这棵树表示了当且仅当它可以被表示为一个点往它的后代路径上的经过所有点的字符串。求有多少个字符串被这颗树表示了。<br>$m$次询问，每一次询问给出了26个字母的大小顺序，求被这棵树表示的字符串中第$K_i$小的字符串。<br>数据范围:$n \le 2.5 \times 10^5, m \le 5 \times 10^4$ | | 可以对这一棵树建出后缀自动机。因为后缀自动机是一个DAG，只需要对后缀自动机拓扑排序然后DP一遍就能求出从每个节点出发有多少条不同的路径，于是就得到了第一问的答案。第二位直接DFS就行了。 | |
| 时间复杂度 | $O(n + m)$ | 空间复杂度 | $O(n)$ |

Problem 87

| 试题名称 | Two Companies | | |
|---|---|---|---|
| 试题来源 | Codechef JUNE 14 | 试题编号 | TWOCOMP |
| 题目大意 | | 算法讨论 | |
| 给定一棵$n$个点的树以及两个带权的树链的集合$A$和$B$，你需要在两个集合中分别选出一个集合使得不存在两条被选出的且属于不同集合的树链相交，使得权值和最大。<br>数据范围:$n \le 10^5, \|A\|, \|B\| \le 700$ | | 这显然是一个经典的最大权闭合图的模型，可以使用网络流解决，建图十分简单，问题在于判断两条树链是否相交。<br>判断是否相交也比较简单，只需要查询LCA深度较大的树链的LCA是否在另一条路径上就可以了。 | |
| 时间复杂度 | $O(maxflow(\|A\| + \|B\|, \|A\|\|B\|) + \|A\|\|B\|logn)$ | 空间复杂度 | $O(\|A\|\|B\| + n)$ |

Problem 88

| 试题名称 | Two Roads | | |
|---|---|---|---|
| 试题来源 | Codechef SEPT 13 | 试题编号 | TWOROADS |
| 题目大意 | | 算法讨论 | |
| 给定$n$个点，选择两条直线，最小化所有节点到这两条直线最近距离的平方和。<br>数据范围:$n \le 100$ | | 两条直线夹角的两个角平分线把平面分成了四个区域，每半条直线控制了一个区域。我们可以枚举第一条角平分线，这条直线一定经过两个点，第二条角平分线和第一条垂直并经过一个点，那么就有$O(n)$种，按照其他点在第一条角平分线投影的位置排序，依次扫描。只需要维护每一个区域的点数,$\sum x$,$\sum y$,$\sum xy$,$\sum x^2$,$\sum y^2$就可以求出答案。 | |
| 时间复杂度 | $O(n^3logn)$ | 空间复杂度 | $O(n)$ |

Problem 89

| 试题名称 | Xor Queries | | |
|---|---|---|---|
| 试题来源 | Codechef JAN 15 | 试题编号 | XRQRS |
| 题目大意 | | 算法讨论 | |
| $n$个操作：<br>1.在末尾插入一个数。<br>2.删去末尾$k$个数。<br>3.询问区间$k$小值。<br>4.询问区间与$x$异或值最大的数。<br>5.询问区间不比$x$大的数的个数<br>数据范围:$n \le 5 \times 10^5$ | | 我们可以用可持久化trie来维护序列，那么就很容易删去元素了。每个节点维护元素个数，查询与x异或最大的数非常容易。可以发现trie很类似于值域线段树，于是把它当作线段树就可以支持45两种询问。 | |
| 时间复杂度 | $O(nlogn)$ | 空间复杂度 | $O(nlogn)$ |

Problem 90

| 试题名称 | Trial of Doom | | |
|---|---|---|---|
| 试题来源 | Codechef JULY 11 | 试题编号 | YALOP |
| 题目大意 | | 算法讨论 | |
| 给定一个$n \times m$的网格，格子为蓝或者红，有$k$个红色格子，你可以在这个网格里走，每当你离开一个格子的时候，这个格子和相邻格子会改变颜色，求是否存在一条左下到右上的路径，使得网格里所有格子都变成了蓝色。<br>数据范围:$T \leq 50$,$n, m \leq 10^9$,$min\ (n, m) \leq 40, k \leq 10000$ | | 令$n \leq m$。<br>若$n = 1$，路径长度和n同奇偶，枚举第一个格子的奇偶，求出其他格子的奇偶，进行判断。<br>否则，对于任意格子集合$S$都存在一条路径，使得$S$里的格子经过奇数次，其他格子偶数次。如果n和m很小，直接解方程就可以了。<br>考虑把所有的红格子都移动到第一列，红格子$(i, j)$等价于红格子$(i-1, j)(i-1, j+1)(i-1, j-1)(i-2, j)$，这样就可以把格子向第一列移动，可以发现第$i$列对第$j$列的影响存在一个周期$C$，那么暴力这个$C$就可以了。然后就可以得到$n$个方程，用线性基判断就行。每一列的情况可以用一个long long存储。 | |
| 时间复杂度 | $O(T(k + nC))$ | 空间复杂度 | $O(k + nC)$ |

以下是Challenge类型试题的做法

Challenge Problem 1

| | | | |
|---|---|---|---|
| 试题名称 | Mushroom Cave | | |
| 试题来源 | Codechef JUNE 11 | 试题编号 | CAVE |
| 题目大意 | | 算法讨论 | |
| 有一个$n \times m$的网格，每个点是空点、火炬或者障碍，从左上到右下找一条经过尽可能多的不同格子的路径，使得连续$k$步以内都要走到有火炬的格子。一个火炬只能用一次，捡到一个火炬就必须用，并把原来的丢掉。<br>数据范围:$1n, m \leq 100, k \leq 15, T \leq 10$ | | 把每个火炬看成关键点，可以先求一个关键点，我们可以首先找到一条关于关键点的尽量长的简单路径，再按照这条路径的顺序依次通过这些关键点，扩展出答案路径。<br>以在原图到终点的最短距离为一个点的估价，先dfs出每次都选择估价大的点的一条路径，再对调整。如果在路径的两个相邻点中还可以再插入一个未出现的点，就插入；如果不能再插入，就将一个点替换成另一个点。多次重复这个过程。<br>然后从前向后每次对相邻两个点都搜出一条当前情况下的最优方案。再从前向后重新搜索。多次重复这个过程。可以在搜索中加入一些随机化。 | |
| 时间复杂度 | 由于有大量搜索部分不好计算。 | 空间复杂度 | $O(n^2m^2)$，但由于数据生成方式远远不到。 |

Challenge Problem 2

| | | | |
|---|---|---|---|
| 试题名称 | To challenge or not | | |
| 试题来源 | Codechef JUNE 13 | 试题编号 | CHAORNOT |
| 题目大意 | | 算法讨论 | |
| 给定一个大小为$n$的集合，求尽量大的子集$B$使得不存在大小为3的等差数列。<br>数据范围:$n \leq 10^5$ | | 考虑直接贪心，把A排序，从小到大依次加入，能加就加。令答案为$k$，那么做一遍的复杂度是$O(n + k^2)$的。而$k$不会很大。<br>为了更优可以加入一些随机化，如果一个数可以加入答案，我们有$p$的概率不选，多次执行这个算法。在我的程序里$p = \frac{1}{100}$，执行了300次。<br>还有一种贪心方法，可以发现只选择三进制不含2的数肯定满足条件，于是可以先选这一部分再做第一部分的贪心。由于A的所有元素加上同一个值不会有影响，于是可以加上1到200多次贪心。 | |
| 时间复杂度 | $O(500(n + k^2))$ | 空间复杂度 | $O(n)$ |

Challenge Problem 3

| | | | |
|---|---|---|---|
| 试题名称 | Closest Points | | |
| 试题来源 | Codechef JUNE 12 | 试题编号 | CLOSEST |
| 题目大意 | | 算法讨论 | |
| 三维空间中有$n$个点，$q$组询问，从给定的$n$个点中找出距离询问点最近的点的编号。<br>数据范围:$n = q = 50000$ | | 这是一个经典问题，可以直接使用KD-tree，建树的时候可以按照方差最大的那一维划分。<br>为了防止TLE，可以设定一个阈值$Max$，每次询问只在KD-tree上运行$Max$个点。我的程序选择了200。 | |
| 时间复杂度 | $O(qn^{\frac{2}{3}})$ | 空间复杂度 | $O(n)$ |

Challenge Problem 4

| | | | |
|---|---|---|---|
| 试题名称 | Kali and Devtas | | |
| 试题来源 | Codechef DEC 14 | 试题编号 | KALKI |
| 题目大意 | | 算法讨论 | |
| 给定平面上$n$个点，求一个生成树，最小化$C_i$的最大值。<br>$C$值的计算：对于每个点，设其在生成树中相邻的最远点的距离为$R$，那么离该点距离为$R$以内的点的$C$值全部加1。<br>数据范围:$n \leq 400$。 | | 生成树中的边尽量小，$R$就会变小，$C_i$被加的次数就会减小，于是可以直接求最小生成树。可以使得每个点的度数尽量小，那么可以设定一个值$K$，使得两个点$u, v$的距离为$d_u d_v K + dist(u, v)$，每次加完边都进行更新。 | |
| 时间复杂度 | $O(n^2 logn)$ | 空间复杂度 | $O(n^2)$ |

Challenge Problem 5

| | | | |
|---|---|---|---|
| 试题名称 | The Great Plain | | |
| 试题来源 | Codechef OCT 11 | 试题编号 | LAND |
| 题目大意 | | 算法讨论 | |
| 给定一个$n \times m$的网格，有一些格子已经填上了数，你需要在其他的格子中填上整数使得代价和尽量小。一对相邻的格子$(u, v)$的到代价是$2^{\lvert A_u - A_v \rvert}$。<br>数据范围:$T \leq 10, n, m \leq 100$ | | 首先可以随机一组初始解，然后考虑随机更新，每次随机一个点$(x, y)$，选择最优解来更新这个点，重复$t$次。我的程序选择了500000次。 | |
| 时间复杂度 | $O(nm + 50t)$ | 空间复杂度 | $O(nm)$ |

Challenge Problem 6

<table>
<tr><td>试题名称</td><td colspan="3">Maximum Sub-rectangle in Matrix</td></tr>
<tr><td>试题来源</td><td>Codechef OCT 12</td><td>试题编号</td><td>MAXRECT</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给出一个$n \times m$的矩阵$A$，选出一个子矩阵，使得子矩阵元素的和尽量大。<br>子矩阵不要求连续，等价于选出的若干行和若干列在所在处相交的元素。<br>数据范围:$200 \le n, m \le 300$</td><td colspan="2">考虑爬山，先随机出行列的初始解（每行列都有0.5的概率选或不选），然后每次随机一行或一列，将它取反，如果答案增加就保留。执行$k$次。我的程序选择$k = 80000000/(n+m)$。<br>执行$t$次爬山算法，我的程序选择了$t = 20$。但这样可能会超时，那么每500次进行判断，如果没有一行或一列取反会更优就可以结束爬山。</td></tr>
<tr><td>时间复杂度</td><td>$O(tk(n+m))$</td><td>空间复杂度</td><td>$O(nm)$</td></tr>
</table>

Challenge Problem 7

<table>
<tr><td>试题名称</td><td colspan="3">Sereja and Sorting 2</td></tr>
<tr><td>试题来源</td><td>Codechef MARCH 14</td><td>试题编号</td><td>SEASORT2</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">一个长度为$n$的数组，可以通过每次翻转数组的一个区间，使其变为升序，(翻转次数+区间长度和$/n$)尽量小。<br>数据范围:$n \le 10000$，不同的数的个数不超过1050。</td><td colspan="2">先考虑求出可行解，只需要从头开始，每次找到最小值并翻转就好了。<br>由于相同的数很多，考虑一个优化，每次找到一定数量的最小值，从后向前依次翻转到头，这个一定数量可以进行调参得到较优的解。</td></tr>
<tr><td>时间复杂度</td><td>$O(n^2)$</td><td>空间复杂度</td><td>$O(n)$</td></tr>
</table>

Challenge Problem 8

<table>
<tr><td>试题名称</td><td colspan="3">Similar Graphs</td></tr>
<tr><td>试题来源</td><td>Codechef APRIL 12</td><td>试题编号</td><td>SIMGRAPH</td></tr>
<tr><td colspan="2">题目大意</td><td colspan="2">算法讨论</td></tr>
<tr><td colspan="2">给出两张$n$个点的图，要求对两张图的点进行重标号，使得相同边数尽可能大。<br>数据范围:$30 \le n \le 75$。</td><td colspan="2">容易发现我们只需要重标号其中一张图就可以了。<br>可以直接使用模拟退火，先随机一组初始解，每次交换两个点的标号，如果比当前解优就替换，不优的话就以一定概率替换。可以多次退火取最优解。</td></tr>
<tr><td>时间复杂度</td><td>$O(Tn)$，T由调参得到。</td><td>空间复杂度</td><td>$O(n^2)$</td></tr>
</table>

Challenge Problem 9

| | | | |
|---|---|---|---|
| 试题名称 | Simultaneous Nim | | |
| 试题来源 | Codechef SEPT 12 | 试题编号 | SIMNIM |
| 题目大意 | | 算法讨论 | |
| 给定$n$个异或和为0的数$A_i$，把它们分成尽可能多组，使得每一组的异或和都为0。<br>数据范围:$n \leq 1000$ | | 可以使用贪心，我们每次分出一个尽量小的组。考虑如何找到这样一个组，我们可以维护一组线性基，那么剩下的每一个数都可以和这些线性基中的几个组成一个组，取最小的那个组。多随机几次，使用不同的线性基取最小值。 | |
| 时间复杂度 | $O(Tn^2)$ | 空间复杂度 | $O(n)$ |

Challenge Problem 10

| | | | |
|---|---|---|---|
| 试题名称 | Stepping Average | | |
| 试题来源 | Codechef NOV 11 | 试题编号 | STEPAVG |
| 题目大意 | | 算法讨论 | |
| 给定$n$个数$A_i$和一个数$K$，每次选择两个数并放入它们的平均数，要求$n-1$次操作后得到的数尽量接近$K$。<br>数据范围:$n = 1000$ | | 每次取出最小值$L$和最大值$R$，设定一个阈值$W \in (0, 0.5)$。<br>如果$K \leq L(1-W)+RW$，令$L$为最后一次操作的数，其他的尽量接近$2K-L$。<br>如果$K \geq R(1-W)+LW$，令$R$为最后一次操作的数，其他的尽量接近$2K-R$。<br>否则直接合并$L$和$R$。 | |
| 时间复杂度 | $O(Tn^2)$ | 空间复杂度 | $O(n)$ |