CODECHEF 试题泛做

晋城一中 赵鋆峰

2016 年 1 月 6 日

第1题

试题编号	Codechef JUN 11
名称	Minesweeper Reversed
题目大意	定义一个扫雷的逆过程，开始时所有的方块都是打开的，有的含“雷”，剩下的不含。有两种关掉方块的方式：可以通过一次点击来关闭一个方块（可以关闭有“雷”的方块），或者点击其他的方块顺便让这个方块关闭。点击一个方块A以后，按照正常扫雷规则，所有原本可能和A同时打开的方块B会全部关闭。求最少操作次数。$n \times m$的期棋盘,T组数据。 数据范围:$n, m, T \leq 50$
算法讨论	显然打开一颗雷单独需要一次。除此之外还有有数字的格子以及空格子。所有由空格子组成的连通块我们称作一个块。显然点击一个空格子会打开整个块，还会打开周边的有数字的格子。打开一个块有两个方式：一种是点击块中的格子，一种是点击周边的格子。容易发现有数字的格子周围最多有两个块，所以我们可以通过一次点开两个块，这样可以少用一次。这样我们对所有块重构一张图，在有数字的格子周围的两个块（如果有的话）之间连边，然后求一下最大匹配。这样可以最大化少用的次数，而剩下的块都必须再来一次。一般图的最大匹配可以用带花树算法解决。
时空复杂度	时间复杂度$O((nm)^3)$，常数很小数据远远达不到这个复杂度 空间复杂度$O(nm)$

第2题

试题编号	Codechef JUN 11
名称	Attack of the Clones

题目大意	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且仅由0和1 组成的数列的集合，B=0,1，我们称n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： Z 是0-保留值函数集合: 满足f(0, ..., 0) = 0; P 是1-保留值函数集合: 满足f(1, ..., 1) = 1; D 是自对偶函数集合: 满足!f$(x_1, ..., x_n)$ = f$(!x_1, ..., !x_n)$; A 是仿射函数集合: 满足如果f$(a_1, ..., c, ..., a_n)$ = f$(a_1, ..., d, ..., a_n)$ 则f$(b_1, ..., c, ..., b_n) = f(b_1, ..., d, ..., b_n)$的函数，在这里c 和d 都在某个位置$i$，并且这个对于任意i, $a_1, ..., a_n, b_1, ...b_n, c, d$ 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n项函数。组合的方式有取交集，并集，差集以及补集，以字符串的形式给出。每个测试数据固定n，给出m种组合。答案对1000003取模。 数据范围:$n, m \leq 100$,字符串长度L≤ 100
算法讨论	容易发现可以把全集分成16(4个集合可以属于或者不属于，所以是$2^4 = 16$) 个不重不漏的部分，只要我们求出每个部分的集合大小，那么最后的组合一定是这16个部分中某些的并集，要知道是哪些只要用一个栈处理一下即可。Z、P、D 集合的情况比较好考虑，A集合的话，实际上我们可以把满足在A集合的函数与$n + 1$ 个数$(a_0, a_1, \ldots, a_n)$ 对应起来其中$a_i = 0/1$，只要令f$(x_1, x_2, \ldots, x_n) = a_0 + a_1x_1 + \cdots + a_nx_n \bmod 2$。这样A集合的大小就是$2^{n+1}$,然后就可以与其他集合进行组合了。
时空复杂度	时间复杂度$O(2^4 logn + mL)$ 空间复杂度$O(L)$

第3题

试题编号	Codechef JUL 11
名称	Billboards
题目大意	在公路上有n个广告牌，大厨会选择一部分广告牌放广告。为了加深路人的影响，大厨希望任意连续的m个广告牌中至少有k个餐馆广告。由于之前一直在亏钱，大厨希望在满足条件的同时使用最少的广告牌。现在大厨想知道有多少种使用广告牌的方案。T组数据。 数据范围:$n \leq 10^9, k \leq m \leq 500, T \leq 300$

算法讨论	用一个长度为n的01串来表示广告牌放的情况。首先考虑$m\|n$的情况，我们把串分成n/m段，令$pos_{i,j}$表示第i段中第j个1所在的位置，那么对任意$x < y, j$一定有$pos_{x,j} \geq pos_{y,j}$。那么我们把$pos_{i,j}$写成一个矩阵，发现这就是一个半标准的杨氏矩阵。方案数为$\prod_{i,j} \frac{r+j-i}{hook_{i,j}}$。其中$hook_{i,j}$表示同一行纵坐标大于等于$(i,j)$ 的或者同一列横坐标大于等于(i,j)的格子数。直接计算是会T的。但我们发现写出来之后大部分分子分母都是可以约分的。真正需要计算的只有km个，这样就可以直接枚举了。然后我们接着考虑$m \nmid n$的情况，如果$n\%m \leq m-k$，那么每一段的前$n\%m$个数肯定都是0。如果$n\%m \geq m-k$，那么每一段的后$m-n\%m$个数肯定都是1。然后我们再用剩下的位置做上面的算法即可。
时空复杂度	时间复杂度$O(Tkmlog(10^9+7))$ 空间复杂度$O(1)$

第4题

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$n \times m$个格子，每个格子可以是蓝色或红色。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。T组数据。 数据范围:$n,m \leq 10^9, min(n,m) \leq 40, k \leq 10^4, T \leq 50$
算法讨论	不妨设$n \leq m$，那么当$n=1$的时候，路径长度的奇偶性一定和n的奇偶性相同。那么我们可以枚举第一个格子的经过次数，解出每一个格子经过次数，然后判断一下即可。当$n>1$的时候，首先我们可以证明我们可以移动到任意一个格子而不改变任何一个格子经过次数的奇偶性，还可以只改变当前格子的经过次数奇偶性而不影响其他格子。这样实际上任意一种不管每一个格子经过的奇偶性是什么我们都可以通过某种移动方式达到此种状态。那么我们只要看是否存在某一种状态使得所有格子都变成蓝色即可。对于(x,y)这个方格，如果它是红色，那么可以操作$(x-1,y)$来使它变成蓝色。这样每个方格都代表了第一列方格的状态，这可以用递推来计算，在同一行中会出现循环，可以用循环节来计算。红色方格的异或值是需要达到的状态，变量就是最后一列的状态，可以通过高斯消元或者线性基来解决。
时空复杂度	时间复杂度$O(T(k+min(n,m) \times len)), len$是循环节的长度 空间复杂度$O(min(n,m) \times len)$

第5题

试题编号	Codechef SEP 11
名称	Short
题目大意	给你两个数n, k，你需要计算满足下列条的(a, b)的数对的个数，满足$n < a < k, n < b < k$，并且$ab - n$ 可以被$(a - n)(b - n)$整除。T组数据。 数据范围：$n \leq 10^5, k \leq 10^{18}, T \leq 5$
算法讨论	首先令$c = a - n, d = b - n$,那么原条件等价于$cd \mid (c + n)(d + n) - n$,且$1 \leq c, d < k - n$。不妨认为$kcd = cd + cn + dn + n^2 - n$,那么$d = \frac{cn+n^2-n}{kc-c-n}$。不妨认为$c \leq d$，那么可以证明$c \leq 3n$。那么我们只要枚举$c$然后枚举$cn + n^2 - n$的约数判断即可。当$c$较大的时候，$k$较小，不妨当$c > 4000$时将枚举因数改为枚举$k$。注意枚举$cn + n^2 - n$的因数时应该用$c + n - 1$和$n$的质因数进行组合。
时空复杂度	时间复杂度$O(Tn \times lim)$, lim为两种方法枚举数量的上限 空间复杂度$O(nlogn)$

第6题

试题编号	Codechef OCT 11
名称	Sine Partition Function
题目大意	求出$\sum_{k_1+k_2+\cdots+k_M=N} \prod_{i=1}^{M} sin(k_i X)$。 数据范围:$M \leq 30, N \leq 10^9, 0 \leq X \leq 6.28$
算法讨论	记要求的式子为$f(M, N)$,用和角公式可以发现 $f(M, N) = f(M - 1, N - 1)sin(X) + f(M, N - 1)2cos(X) - f(M, N - 2)$ 于是就可以用矩阵快速幂来优化转移了。
时空复杂度	时间复杂度$O(M^3 logN)$ 空间复杂度$O(M^3)$

第7题

试题编号	Codechef OCT 11
名称	The Great Plain
题目大意	一个$n \times m$的网格图，每个数都在1到50之间，现在让你把所有空格子填上数，使得所有相邻格子的差k的2^k 的和尽可能小。数据均为随机生成。这是一道challenge题。数据范围:$n, m \leq 100$

算法讨论	一个显然的想法是一开始给所有空格子一个初值，然后一直不断调整每个格子的值使得答案变得更加优。这样我们再通过调整更新的次数，或者更新格子的顺序，甚至可以卡时就可以拿到0.997甚至0.998的分数。但是这样一个格子一个格子更新我们是有可能走进死路的，有可能同时更改几个格子虽然可能对每个格子都不是最优，但是最后的结果却是最优。而我们之前的算法每次都找当前格子的最优值进行选择。这正是爬山算法的局限性，这时候我们就可以模拟退火。每次以适当的概率接受一个比较差的解，并且这个概率随着时间下降，为了使我们的解趋于稳定。还有方法就是每次把这个格子的值由四周的值的最小值往上调整，直到最优。然后我们在一轮更新之后可以随机把一些格子的值-1，这样就可以避免我们走入死路。假设T为我们更新的轮数，那么我们可以通过调整T的大小或者卡时来逐步逼近最优解。
时空复杂度	时间复杂度$O(Tnm)$ 空间复杂度$O(nm)$

第8题

试题编号	Codechef OCT 11
名称	The Baking Business
题目大意	有S个事件，每个事件可能是出售或者查询。 出售会给出产品的细节包括产品编号和大小(大小可能缺省)，出售的地点细节(所在省,城市,地区)(城市和地区可能缺省)以及购买人的性别和年龄。 查询是查询某性别的在某个年龄段的人在某个限制下的购买总数，限制会给出产品细节以及出售地，同样某些地方可能缺省。 数据范围:$S \leq 100000$。有10种产品，每种都有3种不同的大小。有10个省份，每个省份可以被划分为20 个城市，每个城市又可以被划分成5个地区
算法讨论	这其实是一道模拟题。考虑到查询有参数缺省的话不能进行枚举复杂度可能过高,我们考虑在出售的时候就预处理出某个范围的出售总数。 具体我们可以用$sum[product_id][size_id][province_id][city_id][region_id][M/F][age]$表示在此限制下的出售总和，不妨认为所有有效的编号都是正整数，那么当某一维是0的时候则表示其他维不变的情况下，这一维缺省或者等于任何值的出售总和。 这很容易可以在出售的时候预处理出来。注意要枚举产品的所有情况以及地点的所有情况进行组合。 查询因为是区间和，但范围并不大，所以我们可以考虑暴力枚举。也可以在age一维维护一个树状数组，这样查询的复杂度就降下来了，但修改的复杂度就变大了。
时空复杂度	时间复杂度$O(n)$,常数很大 空间复杂度$O(n)$

第9题

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。 数据范围:$p \le 10^12, T \le 5$
算法讨论	首先$a-=p,b-=p$,就可以将问题转化为求$ab\|p(a+b+p)$的对数。我们可以分三部分计算： 1. $p\|a$且$p\|b$，a和b只有$(1,1)(1,2)(2,1)(2,3)(3,3)$5种可能满足题意 2. $p\nmid a$且$p\nmid b$，那么$(a-1)(b-1)\le p+1$,那么如果$a=b$显然$a=b=1$。否则不妨设$a\le b$, 那么$a\le\sqrt{p+1}+1$,如果求出a，那么$b=\frac{a+p}{ak-1}$。令$d=ak-1$，那么需要满足:$p\nmid a, d\|(a+p), a\|(d+1), b>a$。可以枚举$b$和$d$中的较小值来直接判断，这样上界都是$\sqrt{p+1+\sqrt{p+1}}$。 3. $p\|a$或$p\|b$，实际上每一组2中的合法解(a,b)都对应着两组3中的两个合法解$(a,\frac{p(p+a)}{b})(\frac{p(p+b)}{a},b)$。
时空复杂度	时间复杂度$O(T\sqrt{p})$ 空间复杂度$O(1)$

第10题

试题编号	Codechef DEC 11
名称	Hypertrees
题目大意	一个3-超图类似与一个普通的图,只不过其中的边都连接三个点。一个3-超树是一个去掉任意一条边以后都不连通的3-超图，给定n,问有几种含有n个带标号的点的本质不同的3-超树。T组数据。 数据范围:$3\le n\le 17, T\le 15$,保证答案$< 2^{63}$
算法讨论	一张3-超树可以分为若干个点双连通的3-超树。一个点双连通的3-超树每条边都有且只有一个点只在这条边中出现。这样我们把所有这样的点删掉就成了一个普通的点双连通图。于是我们就可以暴力求出这些普通图的个数，然后就可以得到相应的点双连通的3-超树的数目。然后我们再暴力把某些点双联通的3-超树组合成大的3-超树即可。数据范围较小我们打出表来直接交表即可。
时空复杂度	时间复杂度$O(T)$ 空间复杂度$O(1)$

第11题

试题编号	Codechef JAN 12
名称	Card Shuffle
题目大意	给出n张牌，一开始堆成一叠，从上到下牌上的数字分别是$1, 2, \cdots, n$。现在给出m个操作，每次给出A, B, C，要求依次进行以下操作： • 从牌堆顶端拿走A张牌。 • 再从牌堆顶端拿走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。 数据范围:$n, m \leq 10^5$
算法讨论	题目实际上让我们维护一个序列，支持提取一段数，插入一段数，翻转一段数。这都是$Splay \quad Tree$ 的基本操作。
时空复杂度	时间复杂度$O((n+m)logn)$ 空间复杂度$O(n)$

第12题

试题编号	Codechef FEB 12
名称	Find a Subsequence
题目大意	给一个长度为n的数组a和一个12345的排列b。求一个长度为5的子序列，使得其中元素的相对顺序和b一样。T组数据。 数据范围:$n \leq 1000, T \leq 20$
算法讨论	首先离散化。设最后选出来的子序列是$a_{i1}, a_{i2}, a_{i3}, a_{i4}, a_{i5}$。那么我们枚举$i2$ 和$i4$，这样$i1, i3, i5$的大小范围和位置范围都确定了。但大小范围可能会冲突，我们可以贪心。令最大的那个数尽可能大，最小的那个数尽可能小，这样留给中间的数范围更大就更有可能找到解。在位置范围以及大小范围确定的时候找最小/最大的数可以二分+预处理即可。如果有解的话扫一遍得到答案即可。
时空复杂度	时间复杂度$O(Tn^2logn)$ 空间复杂度$O(n^2)$

第13题

试题编号	Codechef MAR 12
名称	Evil Book
题目大意	有n个厨师，第i个厨师有c_i点烹饪力量和m_i点魔法力量。你可以通过在一次烹饪战斗中打败第i 位厨师获得m_i点魔法力量，同时需要付出c_i 点努力。在这场战斗之后，第i位厨师的魔法力量m_i 变为0。而且你可以通过消耗x点魔法力量，选择某个i并把第i位厨师的烹饪力量c_i和魔法力量m_i 都乘以1/3。注意，只有当你拥有至少x点魔法力量时才能请求邪恶之书的帮助。初始时，魔法力量为0。现在请问至少总共需要多少努力才能收集到至少666点魔法力量？ 数据范围:$n \leq 10, c_i, m_i \leq 10^9, 10 \leq x \leq 666$
算法讨论	考虑到数据范围较少，我们可以采用搜索算法。枚举哪些厨师不用魔法，哪些用一次魔法$\cdots$还有若干剪枝，比如当剩下所有的厨师全部打败也不够666的时候就回溯（可行性剪枝），当前消耗的努力已经大于当前最优值的时候回溯（最优性剪枝）。就可以通过此题了。
时空复杂度	时间复杂度O(未知(搜索算法)) 空间复杂度$O(n)$

第14题

试题编号	Codechef APR 12
名称	Substrings on a Tree
题目大意	给出一棵n个节点的树，每个节点上有一个小写字母。定义树的子串为树上从某个点到它的某个子孙的路径上的字母连接而成的字符串。首先输出这棵树有多少个本质不同的子串。接下来有q个询问，每次给出各字母之间的大小关系，要求输出在此大小关系下第k小的子串。 数据范围:$n \leq 2.5^5, q \leq 5 \times 10^4, k \leq 2^{63} - 1$
算法讨论	首先我们考虑序列上的问题，一个字符串有多少个本质不同的子串。这可以很轻松的用后缀自动机解决。给定字母间的大小关系，然后查找第k小的子串也是扫一遍就可以了。考虑把序列转化到树上，我们只要把后缀自动机换成广义后缀自动机即可。
时空复杂度	时间复杂度$O(n + \sum$询问串长) 空间复杂度$O(n)$

第15题

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，第i行第j列的格子颜色为$a_{i,j}$，选它需要的费用为$b_{i,j}, a_{i,j} \in [-1, nm]$。现在需要在这个矩阵里找出一个连通块。要求不能选颜色为-1的格子，且这个连通块至少包含k个不同的正数颜色。在此情况要求费用最小。 数据范围:$n, m \le 15, k \le 7$
算法讨论	首先如果颜色总数$= k$，那么这其实是一个经典的斯坦纳树问题。令$dp_{i,j,s}$表示当前在第i 行第j列的格子上已经选到的颜色集合为s的最小花费。然后两种转移，一种是枚举自己的子集，一种是由相邻的格子的状态转移过来。复杂度是$O(nm3^k)$。然后现在总颜色数可能大于k，我们可以给每种颜色随机一个$1-k$之间的整数，表示对所有颜色重编号。也就是说有些颜色我们视为了一种颜色。我们发现这样只可能使答案变差，而且得到的方案一定是满足题意的。我们不妨来计算这种方法一次就得到正确答案的概率：$\frac{k!k^{tot-k}}{k^k} = \frac{k!}{k^k}$。其中$tot$为矩阵中不同颜色的数目。假设我们进行了$T$次随机，那么我们得不到正确答案的概率就是$1-(1-\frac{k!}{k^k})^T$。可以看出，随机次数越多越可能得到最优解。于是我们可以通过多次提交来确定T的值或者卡时，在没有超过一定时限的情况下一直随机。需要通过各种方式优化自己程序的常数。
时空复杂度	时间复杂度$O(Tnm3^k)$ 空间复杂度$O(nm2^k)$

第16题

试题编号	Codechef APR 12
名称	Similar Graphs
题目大意	给两张n个点的图，请给两张图的点重新编号，使得两张图的相似程度尽可能大。两张图的公共边越多，则相似程度越大。这是一道challenge题。 数据范围:$T = 5, 30 \le n \le 75$,图以一定的方式随机生成。
算法讨论	我们强制令第一张图的编号不变，给第二张图重新编号。每次可以随机两个点，然后比较一下交换它们的编号之后的代价，然后用爬山或者模拟退火的方式进行调整。通过调整参数就可以拿到满分。
时空复杂度	时间复杂度$O(TQn^3)$，Q为调整的次数 空间复杂度$O(n^2)$

第17题

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。如果你打开第i个盒子，有$\frac{P_i}{100}$的概率能获得V_i美元的钱，而有$\frac{1P_i}{100}$的概率能获得一个钻石。现在有m个物品，分别编号为0 到$m1$，第j个物品需要花费恰好C_j美元的钱和D_j个钻石。当你获得了一定量的钱和钻石后，你总会买尽可能多的物品。注意，每个物品只能购买一次。现在请你算出你打开所有的盒子后，期望能够买到的物品个数。T组数据。 数据范围：$n, m \leq 30, T \leq 5$
算法讨论	令$dp_{i,j}$表示获得i个物品且使用不超过j个钻石最少需要花费的钱。这个可以很容易算出来。2^n的暴力复杂度是无法接受的。n的范围容易让我们联想到meet in the middle。考虑把n个盒子分成两部分$an + bn = n$。那么对于后bn个盒子，我们预处理出sta_i存储所有状态中获得钻石数刚好为i的状态的$(money, pro)$，并且按$money$排序，处理出pro的前缀和。那么我们继续枚举前an个盒子的状态，然后设当前状态为(a_money, a_dia, a_pro)，枚举后一部分获得的钻石b_dia，那么如果我们想获得$count$ 个物品，那么b_money必须满足$dp_{count,a_dia+b_dia} \leq a_money + b_money < dp_{count+1,a_dia+b_dia}$。这样满足条件的$b_money$在$sta_{b_dia}$是一段连续的区间，我们二分出这个区间然后直接查询这段的概率和即可。
时空复杂度	时间复杂度$O(bn2^{bn} + m(bn)^2 2^{an})$,取$an = \frac{n}{3}, bn = \frac{2n}{3}$即可通过此题。 空间复杂度$O(2^{bn} + nm)$

第18题

试题编号	Codechef MAY 12
名称	Selling Tickets
题目大意	厨师在一次晚宴上准备了n道丰盛的菜肴，来自世界各地的m位顾客想要购买宴会的门票。每一位顾客都有两道特别喜爱的菜，而只要吃到了至少一道他喜爱的菜，这位顾客就会感到很高兴。当然，每道菜最多只能供应给一位顾客。厨师想要卖出尽可能多的门票，但同时要能够保证，无论哪些顾客购买门票，所有到来的顾客都能感到高兴。现在，厨师想要问你，他最多能够卖多少门票？T组数据。 数据范围：$n \leq 200, m \leq 500, T \leq 15$

算法讨论	我们把n道菜抽象成n个点，然后在每个顾客喜欢的两道菜之间连一条无向边。那么问题可以转化为求这张图中边数恰好比点数多1的连通子图的点数最少是多少。可以看出最优的话一定不存在度数为1的点，所以$\sum_{x\in V} deg(x) = 2\|E\| = 2\|V\| + 2$。所以只会有两种情况：两个点度数为3，其余点度数为2或者一个点度数为4，其余点度数为2。那么这个连通图的结构只有三种： 1. 两个度数为3的点之间有三条不相交的路径 2. 两个不相交的环之间有一条简单路径连接 3. 两个环在一个顶点相交 第一种结构的话我们只要枚举这两个度数为3的点，然后bfs找出3条最短的路径，更新答案。这三条路径可能相交但是我们可以通过调整起点和终点来使得它们不相交。第二种结构我们可以枚举一简单路径上的一个点x，然后从x开始bfs来建出一棵原图的生成树。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。只要拿权值最小的两个环来更新答案即可，可能会相交但是可以转化成结构一或者结构三。第三种结构和第二种一样，只不过是枚举环的交点即可。
时空复杂度	时间复杂度$O(Tn^2m)$ 空间复杂度$O(n+m)$

第19题

试题编号	Codechef JUN 12
名称	Closest Points
题目大意	在三维空间内给出n个点，然后有q个询问，每次给出一个点，询问与它距离最近的点的标号。距离是欧几里得距离。这是一道challenge 数据范围:$n, q \leq 5\times 10^4$
算法讨论	直接对给出的n个点建立$KDtree$然后每个询问都到$KDtree$里面查询即可。
时空复杂度	时间复杂度$O(mn^{\frac{2}{3}})$ 空间复杂度$O(n)$

第20题

试题编号	Codechef JUN 12
名称	Expected Maximum Matching

题目大意	一个二分图，左边有n个点，右边有m个点，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。 数据范围:$n \le 5, m \le 100$
算法讨论	hall定理告诉我们，如果左边存在k个点和右边的大于等于k个点相邻，那么这个图一定存在一个大小为k的匹配。所以我们可以记录左边的每个点集是否满足条件然后每次枚举右边的点可以匹配的左部点的集合，进行状压DP。这样左边一共有2^{2^n}个状态。但是实际上有很多状态是不合法的。我们可以事先通过一个bfs 求出所有状态发现只有4000多个，于是就可以状压DP了。
时空复杂度	时间复杂度$O(m2^n \times state)$,$state$为状态的个数 空间复杂度$O(m \times state)$

第21题

试题编号	Codechef JUN 12
名称	Chefbook
题目大意	给出m个限制x_i, y_i, W_i, L_i, R_i,要求设定n个非负整数P_i和n个非负整数Q_i,满足$L_i \le X_{a_i} - Y_{b_i} + W_i \le R_i$。在合法的情况下最小化$\sum_{i=1}^{m} X_{a_i} - Y_{b_i} + W_i$。要求输出方案。$T$组数据。 数据范围：$n \le 100, m \le n^2, T \le 3$
算法讨论	可以把所有的限制写成$P_i - P_j \le k$的形式，其中$Q_i = P_{n+i}$。然后我们在i和j之间连一条容量为INF，费用为k的边。设i在m个限制中出现了cnt_i次，那么如果$i \le n$，那么在s和i 之间连容量为cnt_i费用为0的边，否则在i和t之间连容量为cnt_i，费用为0的边。这样求最小费用最大流就得到了$\sum_{i=1}^{m} X_{a_i} - Y_{b_i} + W_i$的最小值。考虑方案的话，利用差分约束系统，如果只连初始限制的边我们能得到一组合法解但不一定是最优解。考虑增加一些限制，在求完最小费用最大流的残量网络中，如果某条限制的流量不是0，那么这条限制一定刚好被达到。所以我们把这些限制加上去然后在差分约束系统中就可以求出一组合法的最优解。
时空复杂度	时间复杂度$O(mincostmaxflow(n, m))$ 空间复杂度$O(m)$

第22题

试题编号	Codechef JUL 12
名称	Dynamic GCD
题目大意	给出一个n个点的树，每个点有点权。接下来有m个操作，每次操作是把u到v路径上的所有点的点权加d或者询问u到v路径上所有点权的gcd。 数据范围:$n, m \le 10^5$

算法讨论	先考虑序列上的问题，不妨设这个序列为a，支持区间加，询问区间gcd。直接做并不能用线段树维护，这是因为$gcd(a+c, b+c)! = gcd(a,b)+c$。然而我们发现$gcd(a,b,c) = gcd(a, b-a, c-b)$。所以要求区间$[x,y]$ 的gcd，就等于$gcd(a_x, gcd(b_{x+1}, b_{x+2}, \cdots, b_y))$，其中$b$数组是$a$数组的差分序列。而实际上区间修改，差分序列只会修改两个元素的值。所以差分序列的区间gcd 很好维护。然后再开一个线段树或者树状数组维护a数组每个元素的值即可。把序列转化到树上只要树链剖分即可。
时空复杂度	时间复杂度$O((n+m)log^3 n)$ 空间复杂度$O(n+m)$

第23题

试题编号	Codechef AUG 12
名称	Two Magicians
题目大意	两个人A和B在一张n个点m条边的无向图玩游戏。一开始A在1号点，B在2号点，A和B 都有p点魔力。轮流操作，每次要进行三步： 1. 从当前房间沿着已有边随便走，最终停在某个房间。如果停在对方所在房间，那么游戏结束。此次操作的人赢 2. 加入一条之前没有的边，不能做到的话此次操作的人输 3. 如果还有魔力的话，可以消耗1魔力传送到任何一个点 问先手必胜还是后手必胜。T组数据。 数据范围:$n \le 7777, m \le 10^4, p \le 10^4, T \le 100$
算法讨论	首先当前的博弈状态可以简化为奇偶连通块的个数，AB所在连通块的奇偶性，AB所剩魔力和不会改变当前图连通性的边的数目的奇偶性，然后直接枚举后继状态DP，递推或者记忆化搜索都可。但这样复杂度是$O(n^2p^2)$的。通过观察我们发现有如下结论：最多只会使用一次魔力，偶连通块数目>5的时候DP值不变，奇连通块数目$>$ 8的时候DP值以4为周期编号。这样我们的状态数就非常少了。
时空复杂度	时间复杂度$O(Tn\alpha(n))$ 空间复杂度$O(n)$

第24题

试题编号	Codechef SEP 12

名称	Annual Parade
题目大意	在一个n个点，m条带权边的有向图中，你需要安排若干条路径(s_i,t_i)，路径满足至少包含两个不同的点，而花费由三部分构成： 1.路径s_i到t_i上经过的所有边的边权和，一条边经过多次就计算多次。 2.如果s_i不等于t_i，那么额外付出C。 3.如果某个点没有被任何一条路径经过，仍然额外付出C。 现在给出了k个C值，你需要回答每个C下的最小花费。 数据范围:$N \le 250, M \le 30000, k \le 10000, C \le 10000$,边权$\le 10000$
算法讨论	首先一条路径(s_i,t_i)走的肯定是s_i到t_i的最短路，所以我们就可以先求一遍最短路然后建一个新图。然后我们可以考虑最小路径覆盖。把每个点拆成两个，由s向所有左部点连容量为1，费用为0 的边，所有右部点向t连容量为1，费用为0的边.然后如果存在i到j的最短路的话，i向$j+n$ 连容量为1，费用为i 到j 的最短路长度的边。那么我们发现，对这张图求最小费用最大流。一开始每个点都没有被覆盖，耗费为$n \times C$，可以认为是n条长度为0的路径。然后每次增广要么是连接了两条不相交的路径，要么将一条路径首尾相连，这样实际上我们的耗费都会减少C。所以不妨设这一次增广的费用是V，那么我们实际上需要耗费的是$V-C$。直到增广到$V>C$，这样说明之后再增广我们的费用就会越来越大了，于是就停止增广。所以多次询问的话只要二分出增广到什么时候即可，也可以对询问事先排个序然后做。
时空复杂度	时间复杂度$O(mincostmaxflow(n,n^2)+n^3+klogn)$ 空间复杂度$O(n^2+m)$

第25题

试题编号	Codechef SEP 12
名称	Knight Moving
题目大意	一张无限大的棋盘。一开始有一个格子在$(0,0)$，我们要按下列的移动方式走到(X,Y)：每一步，只能从(u,v)移动到$(u+Ax,v+Ay)$或$(u+Bx,v+By)$。有k个障碍格，跳的过程中不能跳入障碍格。求合法的方案数。 数据范围：所有坐标的绝对值$\le 500, k \le 15$
算法讨论	如果向量$A=(Ax,Ay)$和向量$B=(Bx,By)$线性无关，那么$C=(x,y)$可以被A和B唯一的表示出来，不妨认为$C=aA+bB$。如果不考虑障碍那么方案数就是$\binom{a+b}{a}$，考虑障碍的话我们只要把障碍格排个序然后容斥一下即可。否则如果向量$A=(Ax,Ay)$和向量$B=(Bx,By)$线性相关的话，那么问题就成了一维问题，我们只要暴力DP就可以了。
时空复杂度	时间复杂度$O(k^2 2^k)$ 空间复杂度$O(500^2)$

第26题

试题编号	Codechef NOV 12
名称	Arithmetic Progressions
题目大意	给定一个长度为n的数组A，求有多少对$i,j,k(1 \le i < j < k \le n)$ 满足$A_k - A_j = A_j - A_i$。 数据范围:$n \le 10^5, A_i \le 3 \times 10^4$
算法讨论	令$m = 30000$，我们把这个序列分成$n/size$块，对于三个数在同一块内，可以暴力$size^2$，对于两个数在同一块内可以和上面一起暴力$size^2$，对于中间的数在当前块内，别的两个数不在当前块内将两边卷积$mlog(m)$x 时间复杂度$O(n \times size + n/size \times m \times log(m))$ 可以看出当$size$取$\sqrt{m \times log(m)}$左右可以达到比较满意的时间。
时空复杂度	时间复杂度$O(n\sqrt{mlog(m)})$ 空间复杂度$O(n+m)$

第27题

试题编号	Codechef NOV 12
名称	Martial Arts
题目大意	一个完全二分图，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，他会在知道了匹配之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。 数据范围：$n \le 100, A_{i,j}, B_{i,j} \le 10^{12}$
算法讨论	我们定义每条边的权值$W_{i,j} = A_{i,j} - B_{i,j}$。只考虑最大化价值和：对方会删掉一条边，当且仅当这条边是匹配边中权值最大的且是正值。考虑枚举对方删的这条边，然后把所有边按权值从小到大排序，如果我们强行匹配上这条边，然后求一个最大匹配，最后获得的价值就是除去这条边以外的最大匹配。可以先让所有边的权值为负无穷，然后每次加入一条边，让这条边权值为正无穷，这样相当于强制选这条边，同时维护最大匹配，更新一下答案，然后再修改这条边的权值为它原来的权值，继续考虑下一条边。这样我们需要解决的问题就是如何快速的更改一条边的权值还能维护最大匹配呢。我们使用KM算法中的每个点的$label$ 值。设改变的边所连的点为i和j。我们先将i和它之前的匹配点$mate(i)$在匹配集中删掉，然后维护$label_i$，令$label_i = max(W_{i,j} - label_j)$，再从$i$点出发找一条增广路使匹配完全。这样就可以高效维护最大匹配了。
时空复杂度	时间复杂度$O(n^4)$ 空间复杂度$O(n^2)$

第28题

试题编号	Codechef DEC 12
名称	Different Trips
题目大意	给出一棵n个点的树，每个点的权值是它的度数。两条路径一样当且仅当这两条路径长度且第i个经过的点的权值一样。问有多少条本质不同的从孩子走向祖先的路径。 数据范围:$n \leq 10^5$
算法讨论	如果是在序列上，这就是一个经典的给出一个字符串，问有多少个本质不同的子串问题。可以很方便的用后缀自动机或者后缀数组解决。在树上的话我们可以用广义后缀自动机，只要每次在添加儿子的字符的时候，$last$ 变回父亲加入后的$last$ 即可。数组开不下我们可以用map。
时空复杂度	时间复杂度$O(nlogn)$ 空间复杂度$O(n)$

第29题

试题编号	Codechef JAN 13
名称	A New Door
题目大意	有一个全黑的二维坐标平面。先将四个点$(0,0),(W,0),(0,H),(W,H)$确定的矩形内部涂成白色，再给定n个圆，把每个圆内涂成黑色。求出白色区域的周长，但只有严格在矩形内部的轮廓才被计算在内。 数据范围:$n \leq 1000$
算法讨论	其实题目就是求每个圆在矩形内且不被其他圆覆盖的部分的弧长。我们可以依次处理每个圆，然后依次判断其他的圆是否与它相交，不相交则不考虑，否则把覆盖的弧的范围加入一个列表，最后我们对这个列表求一个并，这是个经典问题，差分之后排序即可。注意一些特殊情况比如说圆在矩形外或者互相包含的情况，还有极角可能跨越π。
时空复杂度	时间复杂度$O(n^2)$ 空间复杂度$O(n)$

第30题

试题编号	Codechef FEB 13
名称	Observing the Tree

题目大意	给一棵n个点树，点上有点权，一开始都是0。现在有m个操作，每个操作是以下的一种： • 对树上X到Y的路径上的点加一个等差数列 • 询问树上X到Y的路径上的点的点权和 • 所有节点的权值返回第X次修改操作之后的状态 强制在线。 数据范围:$n, m \leq 10^5$
算法讨论	给一个序列区间加上一个等差数列很容易用线段树来维护，每个节点打两个标记X, Y表示这个区间加上了一个以X为首项，Y为公差的等差数列即可。标记很容易合并。现在是树上的路径，我们只要树链剖分之后转化为在序列上的问题即可。然后又要求返回历史记录，只要把线段树可持久化即可。
时空复杂度	时间复杂度$O(mlog^2n)$ 空间复杂度$O(mlog^2n)$

第31题

试题编号	Codechef MAR 13
名称	Little Elephant and Colored Coins
题目大意	有N种硬币，每种硬币有个价值V_i和颜色C_i，不同种的颜色可能相同，每种硬币都有无限多个。现在有Q个询问，每次给出一个S，请用已有的硬币恰好拼出价值S，且最大化使用的颜色种数。 数据范围:$1 \leq n \leq 30, 1 \leq V_i \leq 200000, 1 \leq C_i \leq 10^9, 1 \leq Q \leq 200000, 1 \leq S \leq 10^{18}$
算法讨论	首先先不考虑颜色，如何判断某个值能否被达到。这种问题有一个基本的想法，就是选一个基准m （m 是某个硬币的价值）。然后在模m意义下dp，用dp_i表示取到和模$m = i$最少需要多少价值。这样如果$dp_{S\%m} \leq S$,我们就可以加入适当的m从而构造出一组解，否则肯定是不存在方案的。因为是模意义下所以并不好确定dp的顺序，但实际上我们只要找dp值最小的然后更新一圈即可，这样就不会存在后效性。注意考虑价值为x的硬币时，需要转$gcd(x, m)$圈。然后现在考虑上颜色，只要把我们的dp数组再多开一维即可。$dp_{i,j}$ 表示用了j种颜色，达到价值和模$m = i$的时候最少需要多少价值。这样就和前面的做法一样了。
时空复杂度	时间复杂度$O(N^2V)$ 空间复杂度$O(NV)$

第32题

试题编号	Codechef JUN 13
名称	To challenge or not
题目大意	给出n个互不相同的数，要求选出尽可能多的数，使得不存在三个数可以构成等差数列。这是一道challenge题。 数据范围:$n \le 10^5$
算法讨论	此题清橙上数据过弱，直接把数组排序，然后按从小到大能选就选就能拿到300+分。考虑优化的话，我们可以枚举一下最小的数，时间上过不去的话我们可以随机几百个。然后最优解实际上也不一定是能选就选，所以可以以一定概率选能选的。这样应该能得到更优的解。
时空复杂度	时间复杂度$O(n^2)$（最坏，但数据达不到） 空间复杂度$O(n)$

第33题

试题编号	Codechef JUN 13
名称	Count Special Matrices
题目大意	A是一个$n \times n$的整数矩阵，第x行第y个元素记作$A_{x,y}$。 这个矩阵如果满足以下条件就称它是特殊的： • $A_{x,x} = 0 \ for \ 1 \le x \le n$ • $A_{x,y} = A_{y,x} > 0 \ for \ 1 \le x < y \le n$ • $A_{x,y} \le max(A_{x,z}, A_{z,y}) \ for \ 1 \le x, y, z \le n$ • $A_{x,y} \in 1, 2, \ldots, n-2 \ for \ 1 \le x < y \le n$ • 任意$k \in 1, 2, \ldots, n-2$,存在$x, y \in 1, 2, \ldots, n$满足$A_{x,y} = k$ 输入n，求出$n \times n$的特殊矩阵的个数$mod \ 10^9 + 7$。T组数据。 数据范围:$n \le 10^7, T \le 10^5$
算法讨论	可以得到公式 $$ans_n = \frac{n!(n-1)!}{2^{n-1}}(\frac{n}{2} - \frac{2}{3} - \frac{H_{n-1}}{3})$$ 其中$H_n = \sum_{k=1}^{n} \frac{1}{k}$。所以我们只需要预处理$1-n$的阶乘以及逆元然后每次询问$O(1)$回答即可。
时空复杂度	时间复杂度$O(n+T)$ 空间复杂度$O(n)$

第34题

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根棍子组成,且任意相邻的边都不共线。要求输出方案。 数据范围:$n \le 10^3, k \le 10$,棍子长度$\le 10^9$
算法讨论	首先k条给定长度的棍子的能组成一个凸多边形的充要条件是从小到大排序之后$b_k < \sum_{i=1}^{k-1} b_i$。那么如果不存在能构成k 边形的棍子，那么最长的棍子的长度是以指数形式增长的，而所有棍子的长度又是$\le 10^9$的。所以当超过70 根棍子的时候，必定存在两个k边形，那么排序后暴力找两遍即可。然后当$n \le 70$ 的时候，我们可以证明如果有解的话这$2k$根棍子一定是连续的，那么我们枚举起始点和归属情况然后判断即可。
时空复杂度	时间复杂度$O(70\binom{2k}{k} + nlogn)$ 空间复杂度$O(n)$

第35题

试题编号	Codechef AUG 13
名称	Music & Lyrics
题目大意	给定包含二十六个小写字母大写字母和数字，以及“-”的n个单词和m个句子，求每个单词在所有句子里的出现次数。 数据范围:$n \le 500$,每个单词的长度$\le 1000, m \le 100$,每个句子的长度$\le 5 \times 10^4$
算法讨论	一个直接的思路就是把所有句子建成广义后缀自动机然后跑一遍单词即可。但是这样不仅会MLE而且会TLE。考虑对所有单词建立AC自动机，然后这样每个句子能够对所有单词产生的贡献就是：每个前缀能通过$fail$边跳上去的所有点的答案加一。这样我们在每个前缀到达的位置打个标记，最后统一上传即可。
时空复杂度	时间复杂度$O(\sum$ (单词长度+句子长度)) 空间复杂度$O(\sum$ 单词长度)

第36题

试题编号	Codechef AUG 13
名称	Deleting numbers

题目大意	给出一个长度为n的序列，每次可以选择v和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$a_v = a_{v+t} = a_{v+2t} = \cdots = a_{v+kt}$。然后这些数会被删除，之后所有数前移。然后继续选择新的v和t进行删除。直到删除所有的元素。请输出一组合法的v和t。这是一道challenge题。得分与步数有关。 数据范围:$n \leq 10^5, a_i \leq 10^5$
算法讨论	由于清橙上数据以及评分方式的问题，此题直接每次删去数组最末尾的数就可以得到正确...
时空复杂度	时间复杂度$O(n)$ 空间复杂度$O(n)$

第37题

试题编号	Codechef AUG 13
名称	Prime Distance On Tree
题目大意	给定一棵n个点边权为1的树。问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。 数据范围:$n \leq 5 \times 10^4$
算法讨论	树上统计路径可以考虑点分治。令f_i表示长度为i的路径条数，考虑求出所有的f_i，然后直接统计答案就可以了。假设当前正在考虑经过点x的路径，那么每次考虑它的一棵子树，假设g_i表示祖先到当前考虑的子树的子孙路径长度为i的路径条数，h_i表示祖先到之前已经考虑过的所有子树的子孙长度为i的路径条数。那么$f_i += \sum_{j=0}^{i} g_j h_{i-j}$。这里可以用FFT加速。然后再用$g_i$更新$h_i$即可。注意要先选择深度较小的子树进行考虑。
时空复杂度	时间复杂度$O(nlog^2n)$ 空间复杂度$O(n)$

第38题

试题编号	Codechef SEP 13
名称	Room Corner
题目大意	房间是一个边界水平或竖直的多边形。每个90度内角处的内部格子都站着一个小朋友。每次相邻的两个小朋友可以交换位置，速度都是每秒一格。每次给出一对小朋友，询问他们发生相遇的最短时间。询问次数为T，地图是$n \times m$的。 数据范围:$n, m \leq 2500, T \leq 10^4$

算法讨论	我们把相邻的小朋友连一条边，小朋友们一定是沿这些边移动且交换位置，最后一定会形成一个环，我们从某个小朋友出发绕一圈就可以求出环上每一条边的长度，然后统计答案可以枚举小朋友们移动的方向然后二分得到最中间的那个点然后用前缀和算一下答案即可。实现有很多的细节。
时空复杂度	时间复杂度$O(nm + Tlogn)$ 空间复杂度$O(nm)$

第39题

试题编号	Codechef SEP 13
名称	Two Roads
题目大意	给定平面上的n个点，现在要选择两条直线，最小化所有节点到这两条直线上的距离最小值的平方和。保证没有重点和三点共线。 数据范围：$n \leq 100$
算法讨论	考虑有两条不重合直线，整个平面会被这两条直线的两条垂直角平分线分成4个区域。两个相对的区域距离其中一条比较近，另两个区域距离另一条直线比较近。可以发现最优解一定有一条角平分线经过了两个点，所以我们可以枚举两个点的连线作为一条角平分线，然后将所有点按照在这条角平分线的投影长度排序，这样第二条角平分线对于平面起到的划分效果只有$O(n)$种，于是一共只有$O(n^3)$种状态。确定了直线可以用线性回归来得到答案，在枚举第二条角平分线的同时维护两个区域的答案。
时空复杂度	时间复杂度$O(n^3logn)$ 空间复杂度$O(n)$

第40题

试题编号	Codechef NOV 13
名称	Gangsters of Treeland
题目大意	给定一棵n个节点的树，每个节点有一个颜色，一开始全部互不相同。如果一条边两端的节点的颜色不同，那么经过这条边的花费就是1，否则就是0。现在要求支持m个操作，每个操作是将根到某个点之间的所有点都染成一种之前没有出现过的颜色或者询问某个点的子树内的所有点走到根的花费的平均值。 数据范围:$n, m \leq 2^5$

算法讨论	我们可以把这道题目和Link-Cut-Tree联系起来。一条边如果两端颜色相同那么这条边就是实边，否则就是虚边。那么一个点到根的花费就是虚实边切换的次数。而对根到x之间的所有点都染上一种新颜色实际上就是access 操作，虚实边切换的时候会有些子树内的所有点到根的花费$+1/-1$。只要用线段树或者树状数组维护dfs序即可。
时空复杂度	时间复杂度$O(nlogn + mlog^2n)$ 空间复杂度$O(n)$

第41题

试题编号	Codechef DEC 13
名称	Query on a tree VI
题目大意	给出一棵n个点的树，每个点可以是黑色或者白色，一开始都是白色。支持m个操作，每个操作询问某个点所在的连通块的大小或者修改某个点的颜色。连通块就是颜色都相同的点的集合。 数据范围:$n \le 10^5, m \le 10^5$
算法讨论	树上的连通块，我们可以有一种普遍的思路就是以深度最小的那个点作为该连通块的代表元。这样每次我们每次沿着根向上爬，找到最后一个和查询点颜色相同的祖先输出它代表的连通块的大小即可。考虑暴力做法，令dp_i表示i子树内有多少点和i同属一个连通块。这样$dp_i = \sum_{j \in son_i} dp_j + 1$。考虑时刻维护$dp$ 数组。不妨认为x的颜色是白色，然后现在要修改x为黑色。那么x向上直到第一次碰到黑色的点这之间的点的dp值都要减去dp_x，然后它本身的dp值会变成黑色儿子的dp值之和+1。这一步我们并不能暴力统计，所以我们不但要维护dp数组，还要知道与i颜色不同的儿子的dp值之和。这样在修改颜色的时候就很容易得到新的dp 值。然后找第一次碰到的黑/白色点可以用树链剖分/二分倍增+线段树来找，链上加可以树链剖分+ 线段树/树状数组来搞。具体实现的时候我们可以直接用A和B数组分别存储黑色儿子和白色儿子的dp 值之和。加不加1 要看自己是什么颜色。
时空复杂度	时间复杂度$O(nlog^2n)$ 空间复杂度$O(n)$

第42题

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，问是否存在一个长度为n的序列B满足： • 至少存在一个$0 \le i \le n-1$满足$B_i \neq 0$ • 对于任意$0 \le j \le n-1$满足$\sum_{i=0}^{n-1} A_i B_{(i+j)\%n} = 0$ T组数据 数据范围：$n \le 3 \times 10^4, abs(A_i) \le 10^3, T \le 100, \sum n \le 1.5 \times 10^5$
算法讨论	题目相当于给出一个循环矩阵$C_{i,j} = A_{(i+j)\%n}$，问这个矩阵是否满秩。令$f(x) = \sum_{i=0}^{n-1} A_i x^i$，循环矩阵的秩就等于$n - degree(gcd(f(x), x^n - 1))$。$degree$代表最高项次数。所以如果$f(x)$与$x^n - 1$存在度数不为0的公因式，那么这个矩阵就不满秩。分圆多项式$\phi_n(x) = \prod_{0 \le i < n, (i,n)=1}(x - w_n^i)$。那么$x^n - 1 = \prod_{d\|n} \phi_d(x)$。可以发现分圆多项式是不可约的。那么只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$x^d - 1\|f(x) \times \prod_{i\|d, i\ is\ prime}(x^{d/i} - 1)$。注意到一个结论$\sum_{i \ge 0} b_i x^i \ mod\ (x^n - 1) = \sum_{i \ge 0} b_i x^{i\ mod\ n}$,这样我们就可以模拟算出答案了。
时空复杂度	时间复杂度$O(nlogn + Tn \times d(n)), d(n)$为$n$的因数个数 空间复杂度$O(n)$

第43题

试题编号	Codechef JAN 14
名称	Counting D-sets
题目大意	在一个n维空间中，求直径为m的点集的数量.点集的直径是点集中最远的一对点的切比雪夫距离。注意如果两个点集可以通过平移得到，那么我们认为这两个点集是相同的。T组数据。 数据范围:$n \le 1000, m \le 10^9, T \le 10$
算法讨论	可以用直径$\le m$的点集数目$-$直径$\le m-1$的点集数目得到答案，然后为了不重复我们可以强行令每一维坐标都存在一个0。那么所有点就在一个边长为m 的n维立方体中。我们可以用dp_i表示1到i 维坐标都不为存在0，$i+1$到n维坐标存在0的方案数，然后就可以用简单的容斥来得出答案了。
时空复杂度	时间复杂度$O(Tn^2)$ 空间复杂度$O(n)$

第44题

试题编号	Codechef JAN 14

名称	Counting The Important Pairs
题目大意	给出一张n个点m条边的无向连通图，现在要去掉两条边使得图不连通，求方案数。 数据范围:$n \le 10^5, m \le 3 \times 10^5$
算法讨论	首先求出dfs树，然后给每条非树边随机一个权值，而一条树边的权值就是所有覆盖它的非树边的权值的异或和。这样如果选出的两条边能使图不连通，要么是两条边的权值要么相等要么至少存在一条边的权值为0。只要能求出每条边的权值问题就可以解决了。而求每条边的权值，比方一条非树边连接了(x, y)，那么我们可以在y处打一个标记表示y到根的所有边的权值异或上这条边的权值，在x处打一个同样的标记。这样x 上方的边异或了两次相当于没有异或。这样的标记只要倒着上传即可。注意随机的权值要尽可能大，否则两条不同的边相同的几率会很大。
时空复杂度	时间复杂度$O(n+m)$ 空间复杂度$O(n+m)$

第45题

试题编号	Codechef FEB 14
名称	Graph Challenge
题目大意	给你n个点m条边，这n个点的标号是利用dfs得到的dfs序生成的，即存在一种dfs方案使得标号和dfs 序相同。定义x为y的supreme vertex即存在一条路径$v0 = x, v_1, v_2, ..., v_k = y$，满足$x < y$ 且对于所有的$0 < i < k$都有$v_i > y$。定义x为y的superior vertex即满足x是所有y的supreme vertex 中结点编号最小的点。有q个询问每次给出一个点询问每有多少点视其为supervior vertex。 数据范围:$n \le 10^5, m \le 2 \times 10^5, q \le 10^5$
算法讨论	我们可以很简单的还原这棵搜索树。让每个点以它的前驱节点中标号比它小的最大的点为父亲即可。这样superior vertex 实际上就是每个点的半必经点。而求一张Flow Graph的半必经点是个经典问题。可以很轻松的用并查集解决。
时空复杂度	时间复杂度$O(n\alpha(n))$ 空间复杂度$O(n+m)$

第46题

试题编号	Codechef FEB 14
名称	Count on a Treap

题目大意	维护一棵大根Treap，有n个操作，每个操作是下面的一种： 1. 插入给定关键字和权值的点 2. 删除一个给定关键字的点 3. 返回给定两个点的距离 保证任意时刻树中结点关键字和权值都是两两不同的。不会删除当前Treap中不存在的点。 数据范围:$n \le 2 \times 10^5$
算法讨论	一棵Treap的dfs序就是每个点的rank。那么如果给定两个点x和y($x < y$，x和y都是rank)，可以得知x 和y 的lca 就是$[x, y]$中权值最大的。那么我们只需要知道每个点的深度即可。注意到如果a是b的祖先，那么一定有a 是$[a, b]$或者$[b, a]$ 中权值最大的点。这样我们只需要从x开始不断往前跳到比当前权值大的点。这个其实可以用线段树来统计答案。首先线段树的每个节点记录最大值mx_x，再开一个数组f，然后如果这个节点是父亲的左儿子，那么f_x 表示从mx_x所在的位置向右跳直到跳出父亲的管辖范围会跳几次，否则f_x表示从mx_x 所在的位置往左跳直到跳出父亲的管辖范围会跳几次。另外写函数$getl(k, x)$ 表示当前在线段树的k节点，之前的最大值是x，要往左边跳直到跳出k管辖的范围会跳多少次,$getr(k, x)$类似写。这个和f数组可以互相计算。这样我们就解决了这个问题。注意事先离散化即可。
时空复杂度	时间复杂度$O(nlog^2n)$ 空间复杂度$O(n)$

第47题

试题编号	Codechef MAR 14
名称	Chef and Graph Queries
题目大意	有一个n个点m条无向边的图，边从$1 - m$编号。现在有q个询问，每次给出L, R，询问只保留编号在$[L, R]$ 的边的话，这张图有多少个连通块？ 数据范围:$1 \le n, m, q \le 200000$
算法讨论	考虑离线做，然后从小到大枚举右端点。我们把每条边的边权就赋为它的编号。这样对于一个询问$[L, R]$ 只要知道$1 - R$的边的最大生成树中，编号为$[L, R]$之间的有多少条即可。这样我们需要支持加边，维护最大生成树，LCT可以很方便的维护。顺便把在最大生成树中的边标记一下，然后用树状数组查询一下区间和即可。
时空复杂度	时间复杂度$O((n + q)logn)$ 空间复杂度$O(n)$

第48题

试题编号	Codechef MAR 14
名称	The Street
题目大意	要求维护两个长度为n的序列a和b，支持三种操作： • 对a序列区间加上一个等差数列。 • 对b序列区间对一个等差数列取max。 • 询问$a_i + b_i$。 数据范围:$n \leq 10^9, m \leq 3 \times 10^5$
算法讨论	第一个操作是经典问题，我们只要给线段树的每个节点打两个标记x, y表示给这个区间加了一个以x 为首项，y为公差的等差数列即可。标记很容易合并。第二个操作也是一个经典问题。可以认为第二个操作是加入了一些线段，然后查询所有线段与$x = i$的交点纵坐标的最大值。我们考虑给每个节点一个标记线段。那么当一条线段到了一个节点与该节点的标记线段不相交的时候，我们直接取优的作为新的标记，然后返回。否则我们可以判断这两条线段的交点在该节点的左子树还是右子树，不妨设为左子树，A线段在右子树的时候始终在B 上方。那么对于右子树来说一定不会选择B作为最优值，所以把A打在整个子树上，以后右子树的节点上来的时候直接用A 更新ans。而B对左子树还是有贡献的。所以在左子树我们递归插入B线段，这样不断递归下去。这样一次修改是$O(log^2n)$的。所以查询的时候就往上爬然后用途经的所有线段更新ans 即可。空间的话我们只需要开和查询有关的那$O(mlogn)$ 个点就可以保证空间不会超了。
时空复杂度	时间复杂度$O(mlog^2n)$ 空间复杂度$O(mlogn)$

第49题

试题编号	Codechef APR 14
名称	Sereja and Permutation
题目大意	给出一个长度为n的序列a和两个整数k和s，定义$f(a,i) = s - \sum_{t=i}^{j} a_t$，其中$j$ 是使得$\sum_{t=i}^{j} a_t \leq s$的最大整数。要求对$a$重新排列，使得$\sum_{i=1}^{k} f(a,i)$最小。这是一道challenge题目。$T$组数据。 数据范围：$n \leq 2000, a_i \leq 10000, s \leq 10^9$

算法讨论	当$s \geq \sum_{t=1}^{n} a_t$的时候，显然我们将序列从小到大排序是最好的。而且在一般情况下，这个解也算不错的。但是接下来要得到更优的解就有点儿困难。考虑贪心算法。我们先随机产生$k+1$到n的数，那么在选取第1个数到第k个数的时候，我们可以先最小化$f(a,k)$, 在这个基础上最小化$f(a,k-1)$，具体实现只要尽可能往这个位置放满足条件且没有被用过的最大的数即可。
时空复杂度	时间复杂度$O(Tnlogn \times time)$,$time$为贪心的次数 空间复杂度$O(n)$

第50题

试题编号	Codechef MAY 14
名称	Dynamic Trees and Queries
题目大意	给定一个有n个点的有根树，你需要执行m个询问。每个节点有一个key和一个$value$，每个节点以他的key引用。n个节点的key 分别是0到$n-1$。根节点的key 总是0。询问有以下4种类型: 1. 给定一个节点的key，生成一个新节点，作为这个节点的孩子，新节点的key将是从未使用的正整数中最小的一个，$value$ 将给定 2. 给定一个节点的key（称之为A），将以A为根的子树中所有节点的$value$加上给定值 3. 给定一个节点的key（称之为A），将以A为根的子树中所有节点删除。注意删节点不释放key 4. 给定一个节点的key（称之为A），询问以A为根的子树中所有节点的$value$值的和 强制在线。 数据范围:$n, m \leq 5 \times 10^4$
算法讨论	如果只有2和4操作，显然直接在dfs序上建一棵线段树即可。考虑3操作，要删除一棵子树。我们用$Splay$实现删除即可，顺便维护区间和以及修改区间和。再考虑1操作，加入一个点，我们只要在父亲的入点后面插入两个点分别表示新加入点的入点和出点即可。这样就动态地维护了dfs序。注意$Splay$一个点的时候要把所有祖先的标记$pushdown$。
时空复杂度	时间复杂度$O((n+m)logn)$ 空间复杂度$O(n+m)$

第51题

试题编号	Codechef MAY 14
名称	Sereja and Subsegment Increasings
题目大意	给出两个长度为n的序列A和B。在一次操作中，可以选择两个下标i 和$j(1 \le i \le j \le n)$，然后把所有A_i到A_j之间的元素（包含），增加1 之后模4。请问将A数组转换成B，至少要执行多少次操作。T组数据。 数据范围:$n \le 10^5, T \le 10$
算法讨论	首先令$C_i = ((B_i - A_i)\%4 + 4)\%4$，然后再令$C_i = C_{i+1} - C_i$，不考虑模4的话这时候我们的问题就转化成了一开始有一个全是0的序列D，每次可以选择两个下标i 和$j(0 \le i < j \le n)$，令$D_i - 1, D_j + 1$ ，最少多少次能变成C。答案显然是$\sum_{i=0}^{n} max(0, C_i)$。考虑模4的话实际上就是我们可以让$D_i- = 4, D_j+ = 4$而不花费操作次数。所以我们的目的就变成了最小化$\sum_{i=0}^{n} max(0, C_i)$。对于$\|C_i\| \le 1$，操作的话不会更优。每当碰到-2 或者-3的时候，他们配合前面出现过的2和3一起操作会使解更优。所以记录一下之前的2和3的出现次数即可。注意一起操作之后还可能出现新的2和3。
时空复杂度	时间复杂度$O(Tn)$ 空间复杂度$O(n)$

第52题

试题编号	Codechef MAY 14
名称	Sereja and Sorting 2
题目大意	给出一个长度为n的序列a，要求将它从小到大排序，每次可以选择L, R 然后翻转$a_L, a_{L+1}, ,,, a_{R-1}, a_R$。我们认为每次交换的代价是$R - L + 1$。现在要求代价最小化且操作次数不能超过$n$。 数据范围:$n \le 10^4, a_i \le 5000$,不同的数的个数$\le 1050$
算法讨论	我们考虑从第1个位置开始逐渐将序列变成排好序的样子。对于不同位置上相同的一坨数，我们可以每次把后面的翻往前一个位置的右边，这样如果两个位置是相邻的就可以减少一次翻动。这样能拿到99分。注意到代价和区间长度有关，所以我们可以考虑每次只处理相同的lim 个位置。这样区间翻动的长度会小一点。lim可以通过多次提交来确定最优值。
时空复杂度	时间复杂度$O(n^2)$ 空间复杂度$O(n)$

第53题

试题编号	Codechef JUN 14

名称	Two Companies
题目大意	给出一棵N个点的数目，给出M_1条A公司的路径，M_2条B公司的路径。每条路径有一个正权值。请选出一些路径，在满足不同公司的路径不能相交的情况下，总权值最大。 数据范围:$n \leq 10^5, M_1, M_2 \leq 700$
算法讨论	首先，判断两条路径是否相交很好做，脑补一下就可以知道，只要看LCA较深的那个是否出现在了另一条路径上即可。判断某个点是否出现在一条路径上用dfs序很好判断。预处理LCA的话这部分就可以做到O(1)。然后我们发现对于A公司的一条路径a，以及B 公司的一条路径b，如果a和b相交，那我们是不能同时选的。所以这实际上是一个最大点权独立集的问题。因为是二分图，所以可以很容易用最小割来解决。
时空复杂度	时间复杂度$O(maxflow(M_1 + M_2, M_1M_2))$ 空间复杂度$O(M_1M_2)$

第54题

试题编号	Codechef JUN 14
名称	Sereja and Arcs
题目大意	有n个点，坐标分别为$(1, 0), (2, 0), ..., (n, 0)$。每个点有一个颜色，坐标为$(i, 0)$ 的点的颜色为a_i。所有颜色相同的点对之间都有一条颜色为这两个点的颜色圆弧。所有的圆弧圆心在X轴上，且都在第一象限内。问有多少对不同颜色的圆弧相交。 数据范围:$n \leq 10^5, a_i \leq 10^5$
算法讨论	两条弧之间的位置关系只有$AABB, ABBA, ABAB$这三种，直接计算第三种的方案感觉并不好做，所以考虑补集转化，用所有的弧二元组的对数减去第一种和第二种的方案数即可。$AABB$的方案数很好统计，只要记录一下只考虑前i个位置产生的弧的数量即可。考虑计算第二种的方案数。如果A或者B的颜色出现次数超过了$\sqrt{\frac{n}{logn}}$，不妨设为A，那么我们考虑$O(n)$扫一遍整个数组，计算出每个位置左（右）边的该颜色的个数，然后扫一遍就能统计出A在外面的方案，统计A在里面的方案数的时候需要计算组合数，这也很容易。这部分的复杂度是出现次数超过$\sqrt{\frac{n}{logn}}$ 的颜色种数乘上n。所以这部分的复杂度是$O(n\sqrt{nlogn})$。然后如果A和B 的出现次数都$< \sqrt{\frac{n}{logn}}$,实际上这时候弧的数量最多只有$n\sqrt{\frac{n}{logn}}$,所以我们可以用经典的二维数点的方式来计算这部分，复杂度也是$O(n\sqrt{nlogn})$。实际上写的时候细节非常多而且注意去掉同种颜色之间的弧构成的答案。
时空复杂度	时间复杂度$O(n\sqrt{nlogn})$ 空间复杂度$O(n)$

第55题

试题编号	Codechef JUL 14
名称	Sereja and Equality
题目大意	定义两个长度为n的数组A, B相似，当且仅当离散化后这两个数组完全相同。 对于两个排列$P1, P2$，定义函数$F(P1, P2)$等于满足$P1[l...r]$相似于$P2[l...r](1 \le l \le r \le n)$并且$P1[l...r]$ 包含不超过E个逆序对的数对(l, r)的数目。 求当$P1, P2$取遍所有排列的$F(P1P2)$的总和$mod \quad 10^9 + 7$,T组数据。 数据范围:$T \le 10000, n \le 500, E \le 1000000$
算法讨论	考虑枚举区间计算对答案的贡献。首先可以任意选i个，$P1, P2$都是，然后只要$P1$排列方式确定了那么$P2$只要按相对大小一一对应就可以了。剩下的$n - i$ 个元素可以全排列。最后就是求i个元素逆序对$\le E$ 的排列的个数了。这个可以事先预处理出来。
时空复杂度	时间复杂度$O(N^3 + Tn)$（常数很小，不足1） 空间复杂度$O(n^3)$（常数很小，不足1）

第56题

试题编号	Codechef JUL 14
名称	Game of Numbers
题目大意	给出两个长度为n的数组，$A_1, A_2, ..., A_n$和$B_1, B_2, ..., B_n$。让我们维护两个二元组的集合S_1, S_2，初始时集合均为空。每次操作可以选择两个数对$(i, j), (p, q)$，满足(i, j)不在S_1中,(p, q)不在S_2中，$B_j > A_i, B_p < A_q, gcd(A_i, B_j) \neq 1, gcd(A_q, B_p) \neq 1$，且$\gcd(A_q, B_p)$与$\gcd(A_i, B_j)$ 不互质。如果这样的数对存在，就会将$(i, j), (p, q)$分别加入到集合S_1, S_2中。现在请问最多能进行多少次操作。 数据范围:$n \le 400, A_i, B_i \le 10^9$
算法讨论	首先考虑暴力一点的做法，把所有满足条件的(i, j)拎出来新建一个点，由s向它连容量为1的边。把所有满足条件的(p, q)拎出来新建一个点，由它向t连容量为1的边。然后两个数对$(i, j), (p, q)$满足条件，那么在它们之间连容量为1的边。然后跑s到t的最大流。这样显然得到了正确的答案。但这样边数是n^4的，显然无法在规定时间内跑出答案。考虑优化边数，首先gcd相同的左部点可以缩成一个点，右边也是。然后我们只要由$\gcd(a_i, b_j)$向它的所有素因子连边，然后由$\gcd(a_q, b_p)$的所有素因子向$\gcd(a_q, b_p)$连边，这样的图和之前的图是等价的，而边数却降到了$O(n^2 logn)$，可以通过全部测试数据。
时空复杂度	时间复杂度$O(Tmaxflow(n^2, n^2 logn))$ 空间复杂度$O(n^2 logn)$

第57题

试题编号	Codechef AUG 14
名称	Team Sigma and Fibonacci
题目大意	给出n, m，求$(\sum_{x+y+z=n} 6 \times xyz \times fib_x \times fib_y \times fib_z)\ mod\ m$。$T$组数据。 数据范围：$n \le 10^18, m \le 10^5, \sum m \le 10^6$
算法讨论	$A_i = \sum_{x+y+z=n} 6 \times xyz \times fib_x \times fib_y \times fib_z$,可以求出$A$数列的母函数为$6f(x)x^3(x^2+1)^3$，其中$f(x)$的$x^n$的系数为$\frac{1}{3125}[25\binom{n+5}{5}(fib_{n+6}+2fib_{n+5})+150\binom{n+4}{4}fib_{n+5}+5\binom{7}{2}\binom{n+3}{3}(fib_{n+4}+2fib_{n+3})+5\binom{8}{3}\binom{n+2}{2}fib_{n+3}+\binom{9}{4}(n+1)(fib_{n+2}+2fib_{n+1})+\binom{10}{5}fib_{n+1}]$ 这样就可以直接计算答案了。fib数直接用矩阵快速幂计算即可。注意答案让对m取模，但是有除以3125的操作而且m不一定是质数。我们可以让答案先对$3125m$取模，再把最后结果除以3125即可。时限比较紧需要做一些常数优化。
时空复杂度	时间复杂度$O(Tlogn)$ 空间复杂度$O(1)$

第58题

试题编号	Codechef SEP 14
名称	Rectangle Query
题目大意	在二维平面上，有n 次操作，操作有三种：插入一个矩形、删除一个矩形、询问当前有多少个矩形与询问矩形有公共点。 数据范围:$n \le 10^5$
算法讨论	设一个矩形c的左下角为$(X1_c, Y1_c)$，右上角为$(X2_c, Y2_c)$,那么如果矩形a和矩形b有交的充要条件就是$[X1_a \le X2_b][X1_b \le X2_a][Y1_a \le Y2_b][Y1_b \le Y2_a]$。到这一步实际上我们已经可以用数据结构查询满足条件的矩形的个数了。但由于限制较多复杂度也会很高。考虑补集转化，用总数减去不相交的矩形的数目。一个矩形a与b没有公共点，等价于a在2维坐标中至少有1维与m不相交。考虑容斥，用x坐标不相交的个数与y坐标不相交的个数减去2维都不相交的个数，就是矩形b没有公共点的矩形个数。而2维都不相交的矩形可以分成4种情况分别计算。这四种情况都是对x坐标和y坐标分别有一个限制。这样在不考虑删除操作的情况下就成了经典的二维空间数点问题。删除操作实际上我们可以认为在原位置加入了一个权值为-1 的矩形，然后数点变成查询区域内的点的点权和。这样就成功对问题进行了转化。可以用二维数据结构或者分治+数据结构很方便的解决。
时空复杂度	时间复杂度$O(nlog^2n)$ 空间复杂度$O(n)$

第59题

试题编号	Codechef SEP 14
名称	Fibonacci Numbers on Tree
题目大意	给出一棵n个点的树，点上有点权，一开始都是0。要求支持m个操作，每个操作是下面的一种: 1. 将x到y路径上第k的点的权值加上fib_k，其中fib_k是斐波那契数列的第k项 2. 询问x到y路径上所有点的点权和 3. 询问以x为根，y子树内的所有点的点权和 4. 将所有节点的权值返回到第x操作之后的状态 强制在线，答案对10^9+7取模。 数据范围:$n, m \leq 10^5$
算法讨论	首先$fib_k = \frac{1}{\sqrt{5}}((\frac{\sqrt{5}+1}{2})^n - (\frac{\sqrt{5}-1}{2})^n)$, 而且$\sqrt{5}$在模$10^9+9$意义下可以用一个整数来代替。首先我们树链剖分，并且每次先走重边，这样修改操作实际上相当于区间加一个等比数列，注意到如果是任意的等比数列这样是不能做的，但是这题中公比只有四种可能，这样我们可以对每个公比分别处理，这样对线段树的每个节点维护一个tag表示该区间加上了一个以tag为首项的等比数列，标记显然可以合并。这样我们就可以支持修改了。第三个操作是树链剖分基本操作，第二个操作分析一下x和y的关系然后查询子树即可。第四个操作我们把线段树全部可持久化即可。
时空复杂度	时间复杂度$O(mlog^2n)$ 空间复杂度$O(mlogn)$

第60题

试题编号	Codechef OCT 14
名称	Children Trips
题目大意	给出一棵n个点的树，边有长度，长度只会是1或者2。现在有m个询问，每次给出s, f, p。表示有个人的体力为p，要从s走到f。每天至多走p长度的路。但一天结束的时候必须在某个点休息而不能在路中间。询问这个人最少多少天能够到达目的地。 数据范围:$1 \leq n, m \leq 100000, 2 \leq p \leq 2n$

算法讨论	先建立倍增数组$f_{i,j}$表示i的2^j级祖先。表示考虑当$p > \sqrt{n}$ 的时候，答案不会超过$\sqrt{n}$，所以我们就可以暴力走了，每一步只要在倍增数组上二分即可，这部分的复杂度是$O(\sqrt{n}logn)$。当$p \le \sqrt{n}$ 的时候，我们考虑把所有p 相同的放在一块儿考虑。对于每个p，新建一个倍增数组$g_{i,j}$ 表示在体力为p的情况下从i往上走2^j 天能走到哪儿，然后就可以$O(logn)$ 回答每个询问了。
时空复杂度	时间复杂度$O(n\sqrt{n}logn)$ 空间复杂度$O(nlogn)$

第61题

试题编号	Codechef NOV 14
名称	Chef and Churu
题目大意	给一个长度为n个数列a，有n个函数f，其中$f_i = \sum_{j=l_i}^{r_i} a_j$。现在有$T$个操作，每次给出$x$和$y$，把$a_x$ 修改成y 或者查询$\sum_{i=x}^{y} f_i$。 数据范围:$n, T \le 10^5$,任何时刻$a_i \le 10^9$
算法讨论	考虑分块，把n个函数分成m块。考虑时刻维护块内的所有函数的和。当修改a_x的值的时候，我们需要知道每块内计算了多少次a_x，这个我们可以事先在每块内$O(n)$用差分预处理出来。查询的时候整块的直接查，分散的暴力用树状数组求区间和即可。复杂度是$O(nm + T(m + \frac{n}{m}logn))$，可以看出$m$取$\sqrt{nlogn}$较为合适，但这样会MLE，所以我们可以适当缩小$m$的值。
时空复杂度	时间复杂度$O(nm + T(m + \frac{n}{m}logn))$ 空间复杂度$O(nm)$

第62题

试题编号	Codechef NOV 14
名称	Sereja and Order
题目大意	有n个程序，每个程序都需要在两台电脑上分别运行。第i个程序需要在第一台电脑上运行A_i秒，在第二台电脑上运行B_i 秒。一台电脑不能同时运行两个程序，一个程序也不能同时在两台电脑上运行。请问最少需要用多少时间可以让所有程序在两台电脑上都运行过一遍。 数据范围:$n \le 2 \times 10^5, A_i, B_i \le 10^5$

算法讨论	首先答案的下界是$max(\sum A_i, \sum B_i, max(A_i+B_i))$。可以证明这个下界是可以取到的。如果最大值在$A_i+B_i$取到，那么直接把其他程序往里塞即可。否则运行时间和和较大的那台电脑一定是持续不断工作的。我们可以多次随机一个运行程序的排列，然后通过简单的模拟来看看最终的运行时间是否等于这个下界。复杂度是玄学。但从能够通过所有的测试数据来看最优解应该很稠密。
时空复杂度	时间复杂度$O(n\times$玄学$)$ 空间复杂度$O(n)$

第63题

试题编号	Codechef DEC 14
名称	Course Selection
题目大意	有n门课需要学习，给出m天时间。每门课都需要在这m天的某一天修并且一天就能修完。每天修某门课获得的收益可能不一样。有k个限制条件，要求a必须在b 之前修完。现在请合理安排每天修课的内容（每天可以修多门），最大化获得的总收益。 数据范围:$1 \le n, m \le 100, 0 \le k \le 100$
算法讨论	考虑类似$HNOI2013$切糕最小割的做法，我们把每个点拆成$m+1$个点$c_{i,j}$，表示在第j天修第i 门课。然后我们连边$c_{i,j-1}$到$c_{i,j}$容量为在第j天修第i门课的损失（就是修第i门课的最大收益减去这一天的收益），然后连边$(s, c_{i,0}, inf), (c_{i,m}, t, inf)$,表示第0天一定不会被割（也就是第0 天一定还没修），第m 天一定被割了（也就是第m天一定已经修完了）。然后对于限制a，b，连边$(c_{a,i}, c_{b,i+1}, inf)$，表示如果$a$在第$i$天还没修，那么$b$在第$i+1$天一定不能修。然后求出最小割，用每门课的最大收益-最小损失就是我们真正的最大收益。
时空复杂度	时间复杂度$O(maxflow(n, nm))$ 空间复杂度$O(nm)$

第64题

试题编号	Codechef DEC 14
名称	Kali and Devtas
题目大意	给定二维欧氏平面内的n个点，你需要返回这些点的一个生成树，使得C_i的最大值最小。C_i的定义是：对于每个点，设在生成树中与其相邻的点钟最远的点的距离为R，那么以该点为圆心，半径R以内的的点的C_i全部增加1（包括自身）。T组数据。 数据范围:$n \le 400, T \le 100$,$\|$坐标范围$\| \le 2000$
算法讨论	此题清橙上数据过弱，只要输出合法解就能通过。

时空复杂度	时间复杂度$O(Tn)$ 空间复杂度$O(1)$

第65题

试题编号	Codechef JAN 15
名称	Xor Queries
题目大意	给一个初始为空的整数序列，有M个操作，每个操作是以下的一种： • 在序列最后加上一个数x。 • 在区间$[l, r]$中找到数字y，最大化$(x\ xor\ y)$。 • 删除数组最后k个元素。 • 在区间$[l, r]$中，统计小于等于x的元素个数。 • 在区间$[l, r]$中，找到第k小的数。 数据范围:$M \leq 5 \times 10^5$，$1 \leq$ 每个数$\leq 5 \times 10^5$
算法讨论	在数列后面加入一个数，询问区间某个数的$rank$，询问区间第k大，这都是可持久化线段树/主席树的基本操作。而在数列后面加入一个数，查询区间异或某个数的最大值，是可持久化$trie$的基本操作。删除操作只要把指针前移，插入的时候往指针后面插即可。
时空复杂度	时间复杂度$O(Mlog500000)$ 空间复杂度$O(Mlog500000)$

第66题

试题编号	Codechef JAN 15
名称	Ranka
题目大意	在9×9的棋盘上下围棋，回合可以弃权。要求构造一个合法的n步的过程，不能重复出现相同的局面。 数据范围：$n \leq 10^4$
算法讨论	我们可以构造出n步方案，一开始A从一个点$(1, 1)$开始遍历整个棋盘，先往上走，走到顶的时候移到下一列的最下面，继续进行，B始终弃权，然后当剩下$(1, 1)$的前一个格子的时候，B开始放格子，这样会把A的格子全部移除，然后B又开始遍历，剩一个格子的时候A来放,然后一直继续下去... 很容易看出这是不会重复的。而且步数可以达到$9^4 * 2 > 10000$。

时空复杂度	时间复杂度$O(n)$ 空间复杂度$O(1)$

第67题

试题编号	Codechef JAN 15
名称	Sereja and Number Division 2
题目大意	给出一个整数A，和n个整数B_i，要求对A的数位进行重新排列，使得$\sum_{i=1}^{n} A\% B_i$最小。T组数据。这是一道challenge题。 数据范围：$A \le 10^{1000}, n \le 100, B_i \le 10^6, T \le 100$
算法讨论	此题清橙上数据过弱。我们每次随机两个位置交换，然后计算换了是否更优，优的话就交换。计算代价可以预处理一下$d_{i,j} = 10^i \ \% \ B_j$然后就可以$O(n)$计算代价。调整一下随机的次数就可以通过清橙上的测试数据了。
时空复杂度	时间复杂度$O(Tn \times times), times$为随机的次数 空间复杂度$O(n^2)$

第68题

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T，允许前导0，答案对998244353取模。 输入m，对$T = 0, 1, \ldots, m$输出答案。 数据范围:$n \le 10^9$,一部分数据$p \le 50, m \le 500$,一部分数据$p \le 16, m \le 15000$

算法讨论	暴力就是令$dp_{i,j,k}$表示考虑了i位数字，然后这个数$mod\ p = j$，且每位数字和是k的方案数。转移只需要枚举在最后一位加的数字即可，即 $$dp_{i,j,k} = \sum_{(10*x+y)\ mod\ p=j} dp_{i-1,x,k-y}$$ 考虑到n这么大，肯定要加速计算。现在我们已经能从i的情况快速推到$i+1$的情况，如果能从i的情况快速推到$2i$ 的情况我们就可以类似快速幂一样计算了。实际上： $$dp_{2i,j,k} = \sum_{\substack{(10^i x+y)\ mod\ p=j \\ s+t=k}} dp_{i,x,s} \times dp_{i,y,t}$$ 注意这里如果枚举的话就炸了。而第三维是一个卷积。这样我们可以先把$dp_{i,j}$数组DFT，然后 $$dp_{2i,j,k} = \sum_{(10^i x+y)\ mod\ p=j} dp_{i,x,k} \times dp_{i,y,k}$$ 这样我们就可以暴力枚举了，然后再把$dp_{2i,j}$数组IDFT回来。
时空复杂度	时间复杂度$O(((p^2m + pmlogm)logn)$ 空间复杂度$O(n)$

第69题

试题编号	Codechef FEB 15
名称	Payton numbers
题目大意	定义三元组的乘法 def multiply((a1,b1,c1), (a2,b2,c2)): s = (a1a2 + b1b2 + c1c2) + (a1b2 + b1a2) + (c1 + c2) t = floor[s/2] + 16(c1 + c2) - c1c2 A = (t - 2(a1b2 + b1a2) - (a1c2 + c1a2) + 33(a1 + a2)+ (b1b2 - a1a2)) B = (t - 5(a1b2 + b1a2) - (c1b2 + b1c2) + 33(b1 + b2)+ (2b1b2 + 4a1a2)) if s is even: return (A-540,B-540,24) else: return (A-533,B-533,11) 定义zero：若x*任何y=0，则称x是zero 定义单位元，若x*任何y=y，则称x是单位元 定义质数，若x不是zero且不能分解成两个非单位元的乘积，则称x是质数 给定一个三元组，问它是不是质数。T组数据 数据范围:$abs(a), abs(b) \le 10^7, c = 11\ or\ 24, T \le 10^4$

算法讨论	首先有个结论，任意三元组(a, b, c)可以映射到域$Z[\omega]$中$(33-2a-c)+(b-a)\omega$。那么问题转化为了判断域$Z[\omega]$下的某个数是否是质数。然后我们可以发现$(a+b\omega)' = (a+b-b\omega)$。定义$Nx = xx'$，有如下结论 • 如果x不是整数，那么$x$是质数当且仅当$Nx$是质数 • 如果x是整数，那么$x$是质数当且仅当$x$是质数且要么$abs(x) = 2$，要么$abs(x) \neq 11$且$-11$不是$x$模意义下的二次剩余 所以我们直接用Miller_Rabbin素数判别法即可。
时空复杂度	时间复杂度$O(Tlog(10^7))$ 空间复杂度$O(1)$

第70题

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1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sXt的数对。给定m 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。 数据范围:$n \leq 10^5, m \leq 100$任何时刻边权$\leq 10^6$
算法讨论	令f_i表示gcd$=i$的路径条数，g_i表示gcd是i的倍数的路径条数。那么有 $$g_i = \sum_{i\|j} f_i, f_i = \sum_{i\|j} \mu_i g_i$$ 我们现在是要知道f_1，令$T = 1000000$,所以只要求出所有的$i \leq T$的g_i即可。如果没有修改的话，求g_i 只需要把所有边权是i的倍数的边依次加进去然后用并查集就可以求出路径条数。现在有修改的话，我们事先记录下所有和修改有关的边，然后把其他边加进去，依次遍历每个时刻看看这些边现在加不加，考虑完一个时刻后还要复原。所以要用启发式合并的并查集。
时空复杂度	时间复杂度$O(T + m^2klogn)$,其中k是$\leq T$的数中约数最多的数的约数 空间复杂度$O(kn)$

第71题

试题编号	Codechef MAR 15

名称	Random Number Generator
题目大意	给出一个数列的递推公式$A_i = \sum_{j=1}^{k} B_j \times A_{i-j} \ mod\ 104857601$以及该数列的前$k$项，求$A_n$。 数据范围:$n \le 10^{18}, k \le 3 \times 10^4$
算法讨论	用矩阵乘法来加速复杂度是$O(k^3 logn)$的，显然无法接受。但我们可以用矩阵的特征多项式来优化矩阵乘法。这个矩阵的特征多项式是$f(x) = x^k - \sum_{i=1}^{k} B_k x^{k-1}$,令$g(x) = x^n \ mod\ f(x)$,那么答案就是$\sum_{i=1}^{k}[x^{i-1}]g(x) \times B_i$。求$g(x)$我们可以用类似快速幂的方法FFT然后再用多项式除法来进行取模。
时空复杂度	时间复杂度$O(klogklogn)$ 空间复杂度$O(k)$

第72题

试题编号	Codechef APR 15
名称	Black-white Board Game
题目大意	一个$n \times n$的矩阵，第i行的第l_i到r_i个格子是黑色的。其余格子全是白色的。现在有两个人A和B要玩游戏，每人每次要写出一个排列p，要求所有(i, p_i)的格子都是黑色的，且A的排列要求逆序对数为偶数，B要求为奇数。每次写出的排列不能重复。如果某一次某个人写不出满足题意的排列，那么这个人就输了。如果两个人同时写不出，就算平局。现在问你在两个人都非常聪明的情况下，游戏结果是如何的。 数据范围:$n \le 10^5$
算法讨论	我们可以认为黑色格子是1。然后认为一个排列的逆序对数如果是偶数，那么这个排列的权值为1。否则为-1。那么只要计算所有排列的权值之和并且判断正负即可。我们发现所有排列的权值之和实际上就是这个矩阵的行列式。求行列式用高斯消元的$O(n^3)$的。我们需要快速的消元。这个矩阵特殊的就是每一行的1都是连续的。然后我们发现如果我们每次用起点固定，终点较近的那一行去消所有起点和这一行一样的，那么所有行的1还是连续的，并且这些行的起点都变成用来消元的那一行的终点的下一行了。这样我们需要支持快速查找起点相同的行中终点最小的那个，以及合并两个集合。可以用可合并堆达到较好的复杂度。
时空复杂度	时间复杂度$O(nlogn)$ 空间复杂度$O(n)$

第73题

试题编号	Codechef APR 15
名称	Little Party

题目大意	给定一个由m个元素组成的集合，每个元素由n个布尔变量组成，每个变量可以为真或假。要求找出一个总长度最小的基集合，使得以这个基集合中元素为子集的集合的并恰好为给出的集合。基集合中元素由n个变量中某些变量组成，每个变量仍可以为真或假。给出的m个元素一定一定包含所有n个变量。T组数据。 数据范围:$n \le 5, m \le 1000, T \le 120$
算法讨论	首先暴力判断所有的3^n个元素是否可能在基集合中。然后我们采用搜索来得到最优解，需要用可行性剪枝，最优性剪枝以及记忆化搜索才能通过此题。
时空复杂度	时间复杂度$O(T2^{3^n})$,达不到这个上界 空间复杂度$O(3^n + m)$

第74题

试题编号	Codechef MAY 15
名称	Chef and Balanced Strings
题目大意	给出一个由n个小写字母组成的字符串，一个字符串是平衡的当且仅当每种字母出现了偶数次。现在有Q组询问，每次给出l, r，询问 $$\sum_{\substack{l \le s < e \le r \\ T_{s,e} \text{ is balanced}}} \|T_{s,e}\|^{type}$$ 强制在线。 数据范围:$n, Q \le 10^5, 0 \le type \le 2$
算法讨论	首先可以把每个位置考虑成一个26位的二进制数，a_i表示$T_{1,i}$出现的26个字母的次数mod 2的情况，这样如果$[x, y]$平衡当且仅当$a_y = a_{x-1}$。然后我们可以考虑分块。把整个序列分成m块。然后预处理出以每块开头为开头到所有点的答案以及以每块结尾为结尾到所有点的答案。这只要用一个hash或者map记录出现a_i 的次数，i 之和，以及i^2之和即可。然后零碎的直接暴力再扫一遍即可。复杂度$O(nm + Q\frac{n}{m})$，可以看出m取$\sqrt{n}$ 效果较好,但这样空间可能会超，所以我们可以适当减小m的值。
时空复杂度	时间复杂度$O(nm + Q\frac{n}{m})$ 空间复杂度$O(nm)$

第75题

试题编号	Codechef MAY 15
名称	Counting on a directed graph

题目大意	给定一个n个点（从1到n标号）m条边的有向图。请你统计无序对(X,Y)的个数，其中(X,Y)满足存在一条从点1到点X的路径，和一条从点1到点Y的路径，且两条路径除了点1以外没有公共点。 数据范围:$n \le 10^5, m \le 5 \times 10^5$
算法讨论	可以看出(X,Y)满足条件当且仅当在这张图的Dominator Tree中，LCA(X,Y)=1。因此我们建出Dominator Tree 然后扫一遍1的儿子，用子树大小统计答案即可。
时空复杂度	时间复杂度$O(n\alpha(n))$ 空间复杂度$O(n+m)$

第76题

试题编号	Codechef JUL 15
名称	Easy Exam
题目大意	有一个有K面的骰子，每个面上的数字分别是1到K。同时给你两个参数L和$F(0 < L <= K)$。你将这个骰子掷N次。令a_i为掷出数字i的次数。求$\prod_{i=1}^{L} a_i^F$ 的期望值。 数据范围:$N, K \le 10^9, LF \le 20000, F \le 1000$
算法讨论	考虑令$x_{i,j}$表示第i次是否掷到j。那么原式= $\prod_{i=1}^{L}(\sum_{j=1}^{N} x_{j,i})^F$,考虑把这个式子乘出来之后计算每一项的贡献。首先每一项有LF 个元素（可能有相同的）。如果这项里同时有$x_{c,a}$和$x_{c,b}$且a和b不同，那么这一项贡献为0。否则如果出现了tot个不同的c，那么贡献就是$\frac{1}{K^tot}$。那么我们可以先计算a固定的情况，令$dp_{i,j}$表示$(\sum_{c=1}^{N} x_{c,a})^F$表示展开项已经考虑了$i$项，出现了$j$个不同的$c$ 的方案数。那么有$dp_{i,j} = dp_{i-1,j-1} + dp_{i-1,j} \times j$。然后考虑合并$L$个数字。$DP_{n,m}$表示$\prod_{i=1}^{n}(\sum_{j=1}^{N} x_{j,i})^F$的展开项中出现了$m$个不同的$c$的方案数。那么$DP_{n,m} = \sum_{i=1}^{min(m,F)} DP_{n-1,i} dp_{F,m-i} \binom{m}{i}$。这个其实可以用指数生成函数表示，然后用FFT+ 快速幂快速计算。最后统计答案还需要用到lucas定理。
时空复杂度	时间复杂度$O(LFlogLFlogL)$ 空间复杂度$O(LF)$

第77题

试题编号	Codechef JUL 15
名称	A game on a graph

<table>
<tr><td>题目大意</td><td>两个玩家Askar和Bob，正在用一个在n个点m条边的无向图G上的硬币进行游戏。这个游戏如下进行：

Askar选择一个起始点，并把硬币放在这个点上
接下来，两个玩家轮流操作。Bob先进行操作
轮到每个玩家操作时，他需要把硬币沿着一条边移至另一点
硬币不能重复到达同一点
无法操作的玩家将输掉这个游戏

称一个点v为胜利点，当且仅当Askar能够通过选择v为起始点来获胜。假设两个玩家都按最优策略进行操作。给出图G，求出有多少胜利点。T组数据。
数据范围:$n \le 2 \times 10^3, m \le 3 \times 10^5, T \le 3$</td></tr>
<tr><td>算法讨论</td><td>首先点v是胜利点的充要条件是存在一个图G的最大匹配使得v是一个孤立点，即没有被匹配。我们先用带花树算法求一遍最大匹配。我们再从所有孤立点u 找增广路，可以发现每一条长度为偶数的增广路的终点，都可能在最大匹配中成为孤立点，除此之外在带花树花中的所有点都可以通过调整增广路上经过点的顺序使它成为偶数长增广路的终点。因此我们把所有满足要求的点做个标记最后统计答案即可。</td></tr>
<tr><td>时空复杂度</td><td>时间复杂度$O(Tnm)$
空间复杂度$O(n+m)$</td></tr>
</table>