试题泛做

谷晟

1 JULY15.EASYEX Easy Exam

1.1 题目大意

有一个写着 1 到 K 的 K 面骰子，还有两个整数 L 和 F（$0 < L \le K$），掷 N 次骰子，记掷出数 i 的次数为 a_i，求 $a_1^F \times a_2^F \times \cdots \times a_L^F$ 的期望值（设分子为 P，分母为 Q，答案为 $P \times Q^{-1} \mod 2003$，Q 的逆元不存在时输出 0，简化分母为 0 时的情况）。多组数据。

- $1 \le T \le 50$
- $0 < N, K < 10^9$
- $0 < L \times F \le 50000$
- $F \le 1000$

1.2 算法讨论

先讨论 $F = 1$ 的情况，我们把 a_i 拆成 $a_{i,j}$ 表示第 j 回合是否掷出 i。则我们要求 $(a_{1,1}+a_{1,2}+\cdots+a_{1,N})\times(a_{2,1}+a_{2,2}+\cdots+a_{2,N})\times\cdots\times(a_{L,1}+a_{L,2}+\cdots+a_{L,N})$ 的期望。展开后每一项类似 $a_{1,A_1} \times a_{2,A_2} \times \cdots \times a_{L,A_L}$，只要有一组 a_{i,A_i} 全是 1，对答案的贡献就是 $\frac{1}{K^L}$。总贡献可以用组合公式算出，为 $\binom{N}{L} \times L! = \frac{N!}{(N-L)!}$，再除以分母 K^L。

再讨论更一般的 F。此时要求 $(a_{1,1} + a_{1,2} + \cdots + a_{1,N})^F \times (a_{2,1} + a_{2,2} + \cdots + a_{2,N})^F \times \cdots \times (a_{L,1} + a_{L,2} + \cdots + a_{L,N})^F$ 的期望。展开后每一项类似 $a_{1,A_1} \times a_{1,B_1} \times \cdots \times a_{2,A_2} \times a_{2,B_2} \times \cdots$，原来的一个变量变成了 F 个（对于某个 i，$X_i$ 可能等于 Y_i，此时存在两个一样的变量相乘，因为所有变量都是 0 或 1，可以看做一个。但是不允许存在 $X_i = Y_j (i \ne j)$，即某回合掷出两个不同的数的矛盾情况），每一项里不同变量个数由原来的 L 个变成了现在的 L 至 $L \times F$ 个。对于所有有 P 个不同变量的贡献项，总贡献为 $\binom{N}{P} \times P! \times n_p$ 再除以分母 K^P。n_P 表示每种（P 个不同变量已确定）贡献项有几个。

如何计算 n_P？考虑对于某个 i ,$a_{1,A_1} \times a_{1,B_1} \times a_{1,C_1} \times \cdots$ 这 F 个变量中可能有 1 至 F 个不同的变量。有 t 个不同变量的方案数恰好是第二类斯特林数 $S_{F,t}$。由于 $1 \le i \le L$，我们构造这样一个多项式：

$$(S_{F,0}x^0 + S_{F,1}x^1 + \cdots + S_{F,F}x^F)^L$$

，展开后,x^P 的系数就是我们要求的 n_P。可以一并求出所有 n_P。斯特林数可以用 $O(F^2)$ 的递推求。多项式相乘不需要 FFT，注意 $\binom{N}{P} \times P! \times n_p$，$P \ge 2003$ 时模 2003 一定为 0，所以计算多项式最多只要保留 2003 项，可以暴力计算。

1.3 时空复杂度

- 时间：预处理 $O(F^2 + MOD^2)$，单组数据 $O(MOD^2 \log L + L \times F)$；
- 空间：$O(F^2 + MOD^2)$。

2 JULY15.HAMILG A game on a graph

2.1 题目大意

两个玩家 Askar 和 Bob，在无向图 G 上进行游戏。规则：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。多组数据。

- $T \leq 100$
- $N \leq 2000$
- 一个测试点中 M 之和不超过 10^6

2.2 算法讨论

当原图有完备匹配时，先手必败，因为后手总能把硬币移到对应的匹配点上。否则构造一个最大匹配，先手选择任意一个未匹配点为起点，后手移动以后，先手可以把硬币移到对应的匹配点上。后手不可能把硬币移到一个未匹配点，因为这说明发现了一条增广路，而最大匹配不可能有增广路。

为了确定哪些点可以是最大匹配中的未匹配点，可以先任意构造一个最大匹配，从每个未匹配点开始尝试增广，走到的偶点都是可行点（因为此时走过的路径比增广路少一条末端的未匹配边，将这样的“不完整的增广路”反相，匹配数不变，末端的点变成未匹配点）。

原图是一般图，需要使用带花树开花算法求最大匹配。

2.3 时空复杂度

- 时间：$O(NM)$ 优化后可过本题；
- 空间：$O(N + M)$。

3 JUNE15.CHEFBOOK Chefbook

3.1 题目大意

有 N 个人，M 个单向关系。每个关系 $< i, j >$ 有一个数值 L_{ij} 和限制范围 $[S_{ij}, T_{ij}]$，可以调整 P,Q 两个数组来修正关系数值，修正后的关系数值为 $W_{ij} = P_i + L_{ij} - Q_j$，要在满足所有 $S_{ij} \leq W_{ij} \leq T_{ij}$ 的情况下最大化 $\sum W_{ij}$。

- $N \leq 100$。

3.2 算法讨论

本题中对关系数值的约束是一组差分约束，仅利用“差分约束系统”可以求得一组可行解，但无法求最优解。注意到原题是一个线性规划问题，但线性规划通用解法复杂度太高，无法使用。将原问题写成线性规划的标准形式，可以发现它的对偶问题恰好是一个最小费用最大流问题。通过费用流解决对偶问题，再根据“线性规划对偶原理”给原问题添加最优解下的约束条件，此时用“差分约束系统”求出的就是最优解。

3.3 时空复杂度

- $O(\text{min-cost-max-flow}(N, M))$

4 JUNE15.CONPOIN Connect Points

4.1 题目大意

输入一个无向图，判断它是不是最大（不能再加边）平面图。

4.2 算法讨论

最大平面图等价于边数为 $3N-6$ 的平面图。本题可以使用一般的平面图嵌入算法来解决，但非常复杂。这里介绍一种简单的方法。

最大平面图的所有面均为三角形。一个至少有 4 个点的最大平面图不可能有小于等于 2 度的点，且最小度数不超过 5。考虑以下删点操作：

- 删除任意一个 3 度点
- 删除任意一个 4 度点，在四元环上连接一条对角线
- 删除任意一个 5 度点，在五元环上连接两条从一个点引出的“对角线”

对一个最大平面图进行以上任意操作，剩下的还是最大平面图。若原图是最大平面图，不断删点后一定会形成一个 4 点完全图 K_4。如果删点过程中出现意外情况（发现 2 度点；删 4 度或 5 度点时发现意外的边；最后形成的不是 K_4），说明原图一定不是最大平面图。考虑到有少量特殊构造的非平面图也能完成以上删点过程，在删点完成后，逆序将删去的点加回图中，同时检查形成的三角形面是否正确。如果顺利删点再恢复原图，说明原图一定是最大平面图。

4.3 时空复杂度

- 时间：$O(N+M)$；
- 空间：$O(N+M)$。

5 MAY15.CBAL Chef and Balanced Strings

5.1 题目大意

定义平衡字符串为每种字符都出现偶数次的字符串。给定一个只包含小写字母的字符串和若干询问，每个询问给定一个区间和一个数 T（0,1,2），需要回答在这个区间中所有平衡子串的长度的 T 次方之和。要求在线回答询问。$N \leq 100000$。

5.2 算法讨论

首先计算所有字母出现次数的前缀奇偶性，可以表示为一个 2^{26} 以内的整数，离散化后可以再缩小到 N 以内。原来的字符串变成了整数数组，平衡字符串变成了两个相等的整数，子串长度变成了坐标之差。很容易得出一种线性回答一个询问的方法，即扫描相应区间，用数组记录每个整数的出现次数、出现坐标之和、出现坐标平方之和。

然后，将数组按 $\sqrt{N}$ 的大小分块，线性扫描 $2\sqrt{N}$ 次预处理出询问的左端点或右端点在块的边界上时询问的答案。对于一个至少跨过一个整块的一般的询问 $[L,R]$，找出被它包含的最大的由整个的块组成的子区间 $[S,E]$，则 $ANS[L,R] = ANS[L,E] + ANS[S,R] - ANS[S,E] + G[L..S,E..R]$，$G[L..S,E..R]$ 表示第一个数在 $[L,S]$ 范围内，第二个数在 $[E..R]$ 范围内的答案，因为两个区间都在一块以内，线性求解复杂度不超过 $O(\sqrt{N})$，$ANS[L,E], ANS[S,R], ANS[S,E]$ 都已预处理出。

5.3 时空复杂度

- 时间：$O(N^{1.5})$；
- 空间：$O(N^{1.5})$。

6 MAY15.GRAPHCNT Counting on a directed graph

6.1 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。$N \le 100000, M \le 500000$

6.2 算法讨论

对于一个给定起点的有向图，如果从起点到达点 X 的所有路径都经过点 Y，称 Y 为 X 的必经点，所有必经点中离起点最远的称为最近必经点。每个点的最近必经点是唯一的，将每个点和它的最近必经点连起来，得到一棵树，称为 Dominator Tree。计算出 Dominator Tree 以后，根据题意，答案为根节点的所有子树大小两两相乘之和。利用 Lengauer-Tarjan 算法可以快速求出一般有向图的 Dominator Tree。

6.3 时空复杂度

- 时间：$O((N+M)\log N)$；
- 空间：$O(N+M)$。

7 APRIL15.BWGAME Black-white Board Game

7.1 题目大意

给定一个 $N \times N$ 的矩阵，每行有连续一段染成黑色。两个人玩游戏，每回合两人各取一个没取过的排列 P，一个人必须取逆序对数为奇数的排列，另一

个人必须取逆序对数为偶数的排列。排列 P 必须满足对应的格子 (i, P_i) 都是黑色的。谁先取不了谁输。问哪个人会胜利。$N \leq 100000$

7.2 算法讨论

通过行列式的正负，可以比较 01 矩阵中对应格全为 1 的奇排列和偶排列哪个多。但本题矩阵过大，无法直接高斯消元计算行列式。考虑用数据结构加速消元。由于每一行都只有连续一段为 1，把行按照出现 1 的左端点归类。每次取左端点最靠左的那一类中，右端点最靠左的一行 $[L, R_0]$（如果发现有两行一样，不用继续计算，行列式为 0），用它来对同类其它行 $[L, R_x]$ 消元，变为 $[R_0+1, R_x]$，并入另外一类。可以用可并堆维护这些行。

7.3 时空复杂度

- 时间：$O(N \log N)$；
- 空间：$O(N)$。

8 APRIL15.LPARTY Little Party

8.1 题目大意

有 N 个人，M 场派对，这些派对都不太愉快。告诉你每场派对中每个人的心情（好或不好）。一旦出现就会导致不愉快的心情组合称为为基子集，求一组可以总结所有派对的总大小最小的基子集。例如三个派对 AbC, Abc, aBC，Ab, aBC, AbC, Abc 都是合法的基子集。最优答案为 Ab, aBC，总大小为 5。

$M \leq 1000, N \leq 5$

8.2 算法讨论

首先注意 $M \leq 1000$ 的条件是没用的，因为最多 $2^5 = 32$ 种不同的派对。首先枚举所有心情组合，找出所有合法的基子集和他们对应的派对集合。原问题变为最小代价集合覆盖问题，即给定若干集合和对应代价，要求选择一组代价之和最小的集合，使得其并集等于全集。这个问题没有多项式算法，只能通过搜索 + 剪枝完成。

8.3 时空复杂度

- 时间：$O(3^N \times M + 2^M)$ $M \leq 32$，应用多种剪枝优化后可以通过本题；
- 空间：$O(3^N)$。

9 MARCH15.TREECNT2 Counting on a Tree

9.1 题目大意

给出一棵边带权的树，统计最大公因数为 1 的路径条数。中途有 Q 次修改，每次修改一条边的权值，要输出 Q+1 个答案。

- $N \leq 100000$。

- $Q \leq 100$。
- 最大边权 $M \leq 10^6$。

9.2 算法讨论

设 F_i 为每条边都被 i 整除的路径条数。如果求出了 F 数组，可以通过“莫比乌斯反演”求得最终答案。重点在怎么求 F 数组。

一个简单的想法是，对于每个 i，仅取能被 i 整除的边，用并查集连边计算答案。i 最大为 M，最多 $N-1$ 条边，复杂度太大。设边权中不同的质因子个数最多为 c，注意到本题 $c \leq 7$，所以每条边最多参与连边 2^c 次，可以预处理对边分类，每轮计算不用遍历所有边。

本题有修改，但次数少，最多 100 次，受影响的边也最多 100 条。我们把不会被修改的边称为“不变边”，其它的边称为“可变边”。每轮计算先合并“不变边”，再处理“可变边”。

为了处理并查集撤销合并的操作，本题需要使用按秩合并，不路径压缩的并查集。

9.3 时空复杂度

- 时间：$O(M + 2^c \times (N + Q^2) \log N)$；
- 空间：$O(M + 2^c \times (N + Q^2))$。

10 FEB15.DEVLOCK Devu and Locks

10.1 题目大意

求有多少 N 位十进制数是 P 的倍数且每位之和小于等于 M，允许前导 0，答案对 998244353 取模。要求对所有的 $0 \leq M \leq MM$ 计算答案。

- $N \leq 10^9$。
- $P \leq 50, MM \leq 500$ 或 $P \leq 16, MM \leq 15000$。

10.2 算法讨论

为了方便，我们把答案写成多项式形式，F_i 的 j 次项系数表示每位之和（下称“代价”）为 j，模数为 i 的十进制数个数。最终答案就是多项式 F_0 的前 $MM + 1$ 项的所有系数。

首先，十进制数第 i 位（从 0 开始）上的 1 对模数的贡献为 $10^i \mod P$，由于模数只有 P 种，会存在一些数位等价。可以预先求出模数贡献为 k 的数位个数 $cnt[k]$。

对于第 k 类数位，我们构造多项式 $(x^0+x+x^2+x^3+x^4+x^5+x^6+x^7+x^8+x^9)^{cnt[k]}$。展开后，$x^t$ 的系数代表代价为 t，具体模数为 $t \times k \mod P$ 的方案数。根据 $i = t \times k \mod P$ 的值将这个多项式拆成 P 个多项式 G_i，和答案多项式 F_i 进行 P^2 次相乘，合并进答案。本题模数是 998244353，多项式模意义下乘法可以用快速数论变换实现。

10.3 时空复杂度

- 时间：$O(P^2 MM \log MM + P^3 MM)$；
- 空间：$O(P \cdot MM)$。

11 JAN15.RANKA Ranka

11.1 题目大意

在 9×9 的棋盘上下围棋，两个人都由你控制。允许跳过。要求局面（当前轮到谁 + 当前棋盘状态）不能重复。构造一局有 N 步的棋局。

- 数据 1：$N = 5000$
- 数据 2：$N = 10000$

11.2 算法讨论

考虑这样一种下法：黑方不断下棋（保持所有黑子在一个连通块内），白方不断跳过，直到棋盘上只剩一个空格时白方下棋吃掉所有黑子。这样循环 80 次步数超过 6000，已经可以构造 $N = 5000$ 的答案了。此时棋盘上有 80 个白棋，黑方可以下棋吃掉所有白棋，局面恢复到类似于开局时的状态，可以再来一轮，即可构造 $N = 10000$ 的方案。合理安排给白方预留的位置，就可以做到局面不重复。

11.3 时空复杂度

本题可以打表

12 DEC14.DIVIDEN Divide or die

12.1 题目大意

给一个 N 度角，询问是否能尺规作图 N 等分成 N 个 1 度角，如果能要输出方案。

12.2 算法讨论

当 $N \equiv 0 \pmod 3$ 时无解。否则可以作正三角形得到 60 度角，作五角星得到 36 度角，从而作出 3 度角，用 3 度角来等分 N 度角。

12.3 时空复杂度

- 时间：$O(N)$；
- 空间：$O(N)$。

13 NOV14.FNCS Chef and Churu

13.1 题目大意

有一个含 N 个数的数组和 N 个函数，每个函数是数组的一个子区间和。要求支持以下两种操作：单点修改数组元素，询问函数的区间和。

- $N, Q \leq 10^5$。

13.2 算法讨论

考虑对函数分块，预处理出数组里每个数在每块函数中的计入次数，单点修改时可以更新每个块的答案，同时树状数组维护数组前缀和。询问时对于完整的块使用已经计算出的块的答案，多出来的那些函数直接利用树状数组暴力。

13.3 时空复杂度

设块大小为 B

- 预处理 $O(\frac{N^2}{B})$
- 单次修改 $O(\frac{N}{B} + \log N)$
- 单次询问 $O(B \log N)$
- 空间：$O(\frac{N^2}{B})$。

14 OCT14.TRIPS Children Trips

14.1 题目大意

给一棵 N 个点的树，边权为 1 或 2。有 Q 个询问 (S, F, P)，要求从 S 走到 F，每一步可以走若干条边，但边权之和不能超过 P，每一步走尽量远，询问要走几步。

14.2 算法讨论

先考虑 P 比较大的询问，可以暴力分步倍增向上走到 LCA 计算答案。P 比较小时因为步数可能很多，就不能暴力了。此时可以对每个不同的较小的 P，再用一个倍增预处理出向上走若干步到达的点，回答询问时倍增步数而不是边数。可以先对询问排序，避免存储太多预处理信息。P 的阈值可以取 $\sqrt{N}$。

14.3 时空复杂度

- 时间：$O(N^{1.5} \log N)$；
- 空间：$O(N \log N)$。

15 SEPT14.QRECT Rectangle Query

15.1 题目大意

要求实现一个程序支持三种操作：插入一个矩形，删除一个矩形，询问所有的矩形有几个和给定的矩形区域相交。

15.2 算法讨论

先考虑一维情况，询问有几个区间与给定区间相交，只要计算“总区间数 – 在给定区间左侧的右端点数 – 在给定区间右侧的左端点数”，使用树状数组可以轻松实现。推广到二维，只要分别考虑 X 轴和 Y 轴上的投影，但还要加上两个投影都不相交的那些（因为重复减去了）。后者是个带修改的二维数点问题，可以用二维数据结构（例如：树状数组套线段树）或分治 + 排序 + 一维数据结构来实现。

15.3 时空复杂度

- 时间：$O(N\log^2 N)$；
- 空间：$O(N\log^2 N)$ （树状数组套动态开点线段树）。

16 AUG14.SIGFIB Team Sigma and Fibonacci

16.1 题目大意

求 $(\sum_{x+y+z=N} 6xyzf_xf_yf_z) \mod M$，$f_i$ 为 Fibonacci 数列第 i 项。多组询问。$N\le 10^{18}, M\le 10^5, \sum M\le 10^6$

16.2 算法讨论

本题有个简便的做法：打出题目所求数列的前几项的表，借助高斯消元可以发现此数列有一个 12 阶的递推式，可以用矩阵快速幂计算任意项。由于 12×12 的矩阵太大，有大量 M 比较小询问的点可能超时，此时可以利用数列的周期性来优化，周期表可以预先打出一部分。

16.3 时空复杂度

- 时间：单组询问未优化 $O(\log N)$，优化后 $O(\log P)$，P 为周期，**注意有 12^3 的大常数**；
- 空间：一个表和若干个 12×12 的矩阵。

17 JULY14.GNUM Game of Numbers

17.1 题目大意

维护两个二元组的集合 S1,S2，初始时集合均为空。每次操作，他将会选择两个数对 (i,j),(p,q)，满足 (i,j) 不在 S1 中,(p,q) 不在 S2 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，就将 (i,j),(p,q) 分别加入到集合 S1,S2 中。问最多能进行几次操作。

- $N \le 400$。
- $A_i, B_i \le 10^9$。

17.2 算法讨论

首先预处理出所有可行数对和对应的 GCD，题目变为二分图匹配问题，可以用最大流解决，但边数有 $O(N^4)$ 条，无法承受。考虑题目里连边的条件是不互质，可以为每个出现的质因子建一个点作为中转点来连接左右包含这个质因子的点，然后最大流。去除多余点，合并等价点和边后，点数和边数会变得非常少。

17.3 时空复杂度

$O(maxflow(N^2 \log N, N^2 \log N))$，但优化后大部分数据中点数和边数远达不到这么多。

18 JUNE14.TWOCOMP Two Companies

18.1 题目大意

给定一棵树和两组带权的树上路径，要求从两组路径中各选一些路径，选出的路径中不属于同一组的不能相交，求最大权值和。

- 点数 $N \le 10^5$。
- 每组路径数 $M \le 700$。

18.2 算法讨论

先用树链剖分预处理出路径的冲突关系，问题变为二分图最大带权独立集问题，可以用最小割解决。

18.3 时空复杂度

- 时间：$O(N + M^2 \log N + maxflow(M, M^2))$；
- 空间：$O(MN)$。

19 MAY14.ANUDTQ Dynamic Trees and Queries

19.1 题目大意

实现一个程序支持以下 4 个树上操作：增加一个叶子，删除一个子树，让某个子树所有点权加 x，询问某个子树点权和。要求在线。

19.2 算法讨论

由于只有子树操作，可以使用 Splay 来维护 DFS 序或欧拉遍历序。

19.3 时空复杂度

- 时间：$O((N+Q)\log N)$;
- 空间：$O(N)$。

20 MARCH14.GERALD07 Chef and Graph Queries

20.1 题目大意

给一个无向图和一些询问。每次询问只有编号在某个区间内的边连通时图中有几个连通块。

- $N, M, Q \le 200000$。

20.2 算法讨论

使用莫队算法 + 按秩合并的并查集来回答所有询问。由于并查集只能撤销最后几次操作，所以本题和通常的莫队算法细节上有一些区别，需要将左端点所在的块分出来，右端点向右移动时将对应边合并进并查集，左端点所在块里面的边在回答询问时合并，回答完撤销，左端点移出块后要暴力重构。在一个块内的询问直接暴力。

20.3 时空复杂度

由于 N, M, Q 同阶，以下均用 N 代替。

- 时间：$O(N^{1.5}\log N)$;
- 空间：$O(N)$。

21 FEB14.DAGCH Graph Challenge

21.1 题目大意

给定一张从 1 号点能到达所有点的有向图，点已经按照 DFS 序标号。对于两个点 x, y 满足 $x < y$ 且存在一条从 x 到 y 的路径满足路径上除 x, y 以外所有点（可能是 0 个）编号（DFS 时间戳）都比 y 大，则称 x 是 y 的 supreme vertex。其中编号最小的 supreme vertex 称为 superior vertex。询问对于每个点，有几个点以它为 superior vertex。

21.2 算法讨论

本题中的 superior vertex 实际上就是计算 dominator tree 的 Lengauer-Tarjan 算法中提到的 semidominator（半必经点）。套用该算法就可以解决本题。

21.3 时空复杂度

- 时间：$O((N+M)\log N)$;
- 空间：$O(N+M)$。

22 JAN14.CNTDSETS Counting D-sets

22.1 题目大意

计算有多少点集，满足其直径恰好等于 D。点集的直径是点集中最远的一对点的切比雪夫距离。平移后可以重合的点集被认为是等价的。模 10^9+7。

- 维度 $N \le 1000$。
- $D \le 10^9$。

22.2 算法讨论

将原题中 $=D$ 的限制换成 $\le D$，则答案为 $Ans_D - Ans_{D-1}$。我们限制每个点的坐标在 $[0, D]$ 内，且每一维都至少有一个点坐标为 0，这样就满足了题目对点集等价的要求。至少有 k 维坐标不满足要求（没 0）的点集数量为 $\binom{N}{k} \times 2^{D^k(D+1)^{N-k}}$，最终答案可以用容斥法来计算。

22.3 时空复杂度

- 时间：预处理 $O(N^2)$，单组数据 $O(N \log \phi)$；
- 空间：$O(N^2)$。

23 DEC13.QTREE6 Query on a tree VI

23.1 题目大意

给定一棵 n 个节点的树，每个节点有一个颜色 (黑/白)，初始都为黑。维护一种数据结构，支持下列操作：

- 0 u：询问有多少点与点 u 连通。两个点 u 和 v 是连通的，当且仅当 u 到 v 最短路径上的所有点 (包括 u 和 v) 颜色都相同。
- 1 u：切换点 u 的颜色 (黑变白，白变黑)。

23.2 算法讨论

先考虑链上的情况，可以用 Splay 来维护，每个节点里保存区间两端的颜色和同色节点数量。树上的情况可以用 Link/cut tree 来维护，每个节点额外维护与它相连的虚边信息，Access 操作时更新。

23.3 时空复杂度

- 时间：$O(n \log n)$；
- 空间：$O(n)$。

24 NOV13.MONOPLOY Gangsters of Treeland

24.1 题目大意

有 N 个城市连成树结构，每个城市开始时由不同的帮会控制。如果两个相邻城市被同一个帮会控制，距离为 0，否则为 1。实现以下两种操作：有一个新的帮会控制了从某个点到根的所有城市；询问某个子树里所有城市到根的距离平均值。

24.2 算法讨论

题目中的“控制”操作相当于 Link/cut tree 的 Access 操作，距离为 0 和 1 的边相当于实边和虚边。可以用一个 Link/cut tree 来模拟操作，用树链剖分统计答案。

24.3 时空复杂度

- 时间：$O(n \log^2 n)$；
- 空间：$O(n)$。

25 OCT13.FN Fibonacci Number

25.1 题目大意

求 Fibonacci 数列中第一个 $F_n \equiv C \pmod P$ 的项

- $P \mod 10$ 是完全平方数。
- $P \le 2 \times 10^9$，P 是质数。
- 如果答案存在，答案不超过 2×10^9。

25.2 算法讨论

根据题目里对模数的限制，一定存在正整数 x 满足 $x^2 \equiv 5 \pmod P$，因此可以在本题中使用 Fibonacci 数列的通项公式。根据通项公式列出方程，可以得到一个关于 ϕ^n 的二次方程，求解此方程后再用 Baby-step-giant-step 算法求解离散对数问题，解出 n 的值。此题中涉及模意义下的平方根的求解，可以使用专门的算法，也可以使用指标（离散对数）法。

25.3 时空复杂度

- 时间：单组数据 $O(\sqrt{P})$；
- 空间：$O(\sqrt{P})$。

26 AUG13.LYRC Music & Lyrics

26.1 题目大意

给一些模板串和文本串，求每个模板串在所有文本串中的匹配次数。

26.2 算法讨论

本题是一个简单的字符串多模匹配问题，直接使用 AC 自动机就可以解决。注意在匹配时不要顺着 Fail 边更新答案，而是先在点上打标记，最后再根据拓扑序统计答案。

26.3 时空复杂度

- 时间：$O(|S|)$；
- 空间：$O(|S|)$。

27 JUNE13.TKCONVEX Two k-Convex Polygons

27.1 题目大意

给 N 根不同长度的棍子，要求回答是否能用这些棍子组成两个凸 k 边形，要输出方案。$n \leq 1000, k \leq 10$，棍子长度不超过 10^9。

27.2 算法讨论

k 根棍子能组成多边形的条件是最长边长度小于其它边之和，能组成多边形就一定能组成凸多边形。本题 k 小，如果无解，最长边的长度会随 N 指数级增长，而本题棍子长度有限，所以 N 足够大时一定有解。本题中 $N > 70$ 则一定有解。先将棍子按长度排序。要拼成多边形，最长边尽量要短。如果有解，一定能找到一种方案，使得选中的 $2k$ 根棍子要么是两段连续 k 根（一段一个凸 k 边形），要么是连续 2k 根。可以很方便地枚举求解。

27.3 时空复杂度

- 时间：$O(M \times \binom{2k}{k} + N \log N), M = \min\{N, 70\}$；
- 空间：$O(N)$。

28 MAY13.QTREE Queries on tree again!

28.1 题目大意

有一个 N 个点 N 条边的无向简单连通图，且保证图中唯一的环长度为奇数。要求支持以下两种操作：对两个点之间的最短路上边权取相反数，询问一条最短路上最大连续边权和（此处最短路不考虑边权）。

28.2 算法讨论

如果是链上问题，可以很方便地用线段树做，每个节点维护权值和，子区间最大值，前缀最大值，后缀最大值。修改操作可以打标记。树上问题可以用树链剖分。本题再多一条边，可以用基环 + 外向树的形式来处理，也可以用生成树 + 一条额外边的形式来处理。

28.3 时空复杂度

- 时间：$O(N\log^2 N)$；
- 空间：$O(N)$。

29 MARCH13.LECOINS Little Elephant and Colored Coins

29.1 题目大意

有 N 种类型的硬币，编号从 1 到，第 i 种硬币价值 V_i 美元，颜色是 C_i，每种硬币都有无限个。Q 个询问，每次询问用这些硬币组成 S 美元最多可以用几种颜色，无解输出-1。

- $N \leq 30$。
- $V_i, Q \leq 200000$。
- $S \leq 10^{18}$。

29.2 算法讨论

先不考虑颜色，只考虑可行性。设最小面值为 M，F_i 表示用其它硬币能凑成的最小的模 M 余 i 的价格。对于一个询问，如果 $F_{S \bmod M} < S$ 则有解且要用到面值 M 的硬币，$F_{S \bmod M} = S$ 则有解且不用面值 M 的硬币，$F_{S \bmod M} > S$ 则无解。数组 F 可以用 SPFA 算法计算。再考虑有颜色的情况。我们把一个方案所用的硬币分成两部分，第一部分由若干不同面值和颜色的硬币组成，决定了颜色数，第二部分硬币不影响颜色数，只用来凑钱。第一部分可用 DP 求解，DP_{ij} 表示用前 i 种颜色的硬币，每种颜色最多拿一个（因为这部分不是用来凑钱的，只决定颜色，所以尽量少拿），总价值为 j 时最多颜色数。第二部分以之前的 DP 为基础，用最开始讲的方法求解，要用到 SPFA，需要加一些优化才能不 TLE。最终算出数组 F，F_{ji} 表示使用 j 种颜色时，能凑成的最小模 M 余 i 的价格。然后就可以回答所有询问了。

Codechef 官方题解里讲述了另一种做法，将状态转移中遇到的环有依据地拆开，避免了 SPFA，在最坏情况下复杂度比本文算法更优。

29.3 时空复杂度

- 时间：DP 为 $O(N\sum V)$，计算 F 数组为 $O(N \times SPFA(M, NM))$，回答一个询问为 $O(N)$；
- 空间：$O(N\sum V)$。

30 FEB13.ROC Room Corner

30.1 题目大意

给一个 $N \times M$ 的 ASCII 字符画成的四连通房间地图，每个房间角落上都有一个小朋友（每行最多两个），将房间墙壁看成一个环，相邻的小朋友可以相互靠近，交换位置，走一格消耗的时间为 1。每次询问一对小朋友相遇至少要多少时间。

30.2 算法讨论

本题重点在如何处理地图，实现时要注意细节（比如可能出现的阶梯状等特殊形状房间）。处理完地图后本题就变成了简单题。假设询问的两个小朋友位置是 A, B，他们在小朋友 C, D 之间相遇，则答案为 $max(|A-C|, |B-D|)+\frac{|C-D|}{2}$，设 P 为 CD 中点，答案就是 $max(|A-P|, |B-P|)$，可以将每个区间的中点插入数组，二分确定最近的中点。注意题目里是个环，可以往两个方向走，一种比较简单的做法是将 N 个位置在后面复制一遍。

30.3 时空复杂度

- 时间：预处理 $O(NM)$，单个询问 $O(N \log N)$；
- 空间：$O(NM)$。

31 JAN13.ANDOOR A New Door

31.1 题目大意

在一个 $W \times H$ 的门上挖 N 个圆洞装玻璃，问玻璃在门内的周长之和，$N \leq 5000$。

31.2 算法讨论

一个圆的有效周长是它没有被其它圆覆盖且不在矩形外面的部分。可以对每个圆，枚举计算被覆盖的圆弧区间（注意这个区间可能跨过-pi/pi 这个点，这时需要拆成两个），再排序扫描计算出未被覆盖的圆弧长度。

31.3 时空复杂度

- 时间：$O(N^2 \log N)$；
- 空间：$O(N)$。

32 DEC12.DIFTRIP Different Trips

32.1 题目大意

N 个城市组成一棵树。假如两个城市的度数相同，则他们被认为是相似的。两条路径被认为是相似的当且仅当他们的长度相同且按顺序一一对应的城市都相似。求有多少种互不相似的祖先-子孙路径。

32.2 算法讨论

把度数看成字符，则互不相似的祖先-子孙路径数量为 Trie 上本质不同的子串数量。可以用后缀自动机解决。

32.3 时空复杂度

$O(N)$

33 NOV12.COUNTARI Arithmetic Progressions

33.1 题目大意

给定 N 个数，问有多少三元组 $(i,j,k), i<j<k$ 满足 A_i, A_j, A_k 为等差数列。

- $N \le 100000$。
- $A_i \le 30000$。

33.2 算法讨论

考虑分块。合法三元组有三种情况：三个点都在一块内，两个在一块内，在三个不同的块中。前两种情况可以用桶排 + 枚举来统计，第三种情况用桶排 + 卷积来统计，卷积可用 FFT 实现。

33.3 时空复杂度

设 B 为块大小。

- 时间：$O(NB + \frac{NA\log A}{B})$；
- 空间：$O(N+A)$。

34 SEPT12.KNGHTMOV Knight Moving

34.1 题目大意

有一张无限大的方格棋盘。有一个“骑士”，它必须从 (0,0) 格开始，按照如下规则，移动至 (X,Y) 格：每一步，它只能从 (u,v) 格移动至 (u+Ax,v+Ay) 或者 (u+Bx,v+By)（不能往回走）。此外，棋盘上有 K 个障碍格，骑士不能进入这些格子。任务是计算骑士有多少种到达指定位置的方案（可能为正无穷）。我们认为两种方案不同，当且仅当它们的步数不同，或者存在某个 i 使得两种方案中，骑士在第 i 步到达的格子不同。所有坐标绝对值不超过 500，不超过 5 组数据，$K \le 15$。

34.2 算法讨论

本题有两种情况。当两个向量线性无关时，可以以这两个向量为基底建立新的坐标系，将所有点映射到新坐标。由于不能往负方向走，转移不可能有环。假设终点的新坐标为 (p,q)，则方案数为 $\binom{p+q}{p}$，对于障碍格，可以根据拓扑序容斥计算。

当两个向量共线时，转化为一维问题，但可能有环。我们可以用一个比较宽松的范围（比如 500^2）把无限棋盘变成有限棋盘（需要使用无穷远点的情况答案一定是正无穷，可以通过找环来判断），然后使用 BFS 标记出可达点判断有无解，用拓扑排序找环和 DP 计算答案。

34.3 时空复杂度

设最大坐标为 M。

- 时间：预处理 $O(M^2)$，二维情况 $O(K^2 \log \phi)$，一维情况 $O(M^2)$
- 空间：$O(M^2)$。

35 AUG12.MAGIC Two Magicians

35.1 题目大意

两个人在一个简单无向图上玩游戏，每个人每回合这样行动：首先走到同一个连通块内的任意点上，如果走到对方所在点，直接获胜，游戏结束；否则连接一条不存在的边；连完边后可以选择消耗 1 点魔法值传送到任意位置（初始两个人的魔法值都为 P）。给定初始状态，询问先手必败还是必胜。$N \leq 7777$。

35.2 算法讨论

表示一个状态需要这些信息：两个玩家所在连通块大小，所有无人连通块大小，连接后不改变连通性的边（下称“无用边”）的条数，魔法值。

考虑“连接一条无用边”这个操作，它只改变这类边的数量，不改变其它状态，可以将其视为有次数限制的“跳过当前回合”操作，只用关心其奇偶性。状态简化为：两个玩家所在连通块的奇偶性，奇数和偶数大小的无人连通块个数，无用边奇偶性，魔法值。状态数 $O(N^2P^2)$，太多。

考虑传送操作，对于其它条件相同的状态，魔法值显然是越多越好（太多了可以不用）。经过观察可以发现，仅有一种情况——所有无人连通块大小为偶数（且至少有一个），两个有人连通块大小为奇数，必须将自己和对方所在连通块连起来，否则必败——必须使用传送，其它所有可能使用传送的情况均可找出一种不用传送的等价替代方案。因为这种情况连完之后所有连通块大小都变成偶数，不可能再满足相同条件，所以传送最多用一次。状态中“魔法值”可以简化为“是否可以传送”。状态数最多 $16N^2$，还是有点多。

打表观察可以发现，当偶数连通块足够多（大于 10 即可）时，答案不变，奇数连通块足够多（大于 8 即可）时，答案以 4 为周期。只有常数个状态，可以轻松解决。

35.3 时空复杂度

预处理和回答一个询问都为 $O(1)$

36 MAY12.LEBOXES Little Elephant and Boxes

36.1 题目大意

有 N 个盒子，打开第 i 个盒子有 P_i 概率获得 V_i 美元，否则获得 1 个钻石。有 M 个物品，每个物品都要花费一定钱和钻石才能购买。问期望能买几个物品。$N, M \leq 30, V_i \leq 10^7$

36.2 算法讨论

考虑用 meet in the middle 策略。把 N 分为 A+B，对后 B 个盒子，暴力预处理出有 i 个钻石时每种可能的钱数和概率，按钱数排序，概率求前缀和。DP 出有 i 个钻石要买 j 个物品最少要多少钱。然后对前 A 个盒子枚举结果，在后 B 个盒子的结果中二分统计答案。

36.3 时空复杂度

- 时间：$O(NM + 2^A B + 2^B)$；
- 空间：$O(NM + 2^B)$。

37 APRIL12.TSUBSTR Substrings on a Tree

37.1 题目大意

有一棵 Trie，首先询问 Trie 中有多少个本质不同的子串。然后 Q 个询问，每次询问第 K 小的子串。

37.2 算法讨论

题目中的两个询问，都是后缀自动机的经典用法。

37.3 时空复杂度

- 时间：$O(N)$；
- 空间：$O(N)$。

38 JULY11.BB Billboards

38.1 题目大意

有 N 个位置可以放广告牌，每连续 M 个位置至少要有 K 个。放置能满足要求的最少的广告牌。求方案数。

- 300 组数据。
- $N \le 10^9$。
- $M \le 500$。

38.2 算法讨论

先考虑 M 整除 N 的情况。依次把连续 M 个位置分成一组。可以证明每组中只需 K 个广告牌。把放置方案写成矩阵形式，A_{ji} 表示第 j 组中第 i 个广告牌的相对位置。可以发现，合法方案对应的矩阵一定满足性质：向下严格单调递增，向右不严格单调递减。满足性质矩阵对应的方案也都合法。设 $T = \frac{N}{M}$，矩阵方案数可以用公式 $\prod \frac{M+j-i}{K+T-i-j+1}$ 来计算。但是矩阵可能很大。观察上述公式，发现分子和分母里有很多项可以约分，最后的有效项数规模不超过 $O(MK)$。M 不整除 N 的情况，在最优解的条件下，会存在一些位置必须放置或不放置广告牌，最终可以将其转换为 M 整除 N 的能解情况。

38.3 时空复杂度

- 时间：$O(MK)$；
- 空间：$O(1)$。

39 MAY15.RNG Random Number Generator

39.1 题目大意

已知一个递推数列的 K 阶递推式和前 K 项，求第 N 项。$K \leq 30000$

39.2 算法讨论

由于 K 太大，不能使用矩阵乘法。本题需要使用特征多项式来优化线性递推式的计算，是一个经典问题。

39.3 时空复杂度

- 时间：$O(K \log K \log N)$；
- 空间：$O(K)$。

40 AUG14.PUSHFLOW Push the Flow!

40.1 题目大意

给定一个每个点最多属于一个环的简单无向图（点-仙人掌），Q 个询问，每次修改一条边的权值，或询问两个点 S, T 的最大流。

40.2 算法讨论

最大流相当于最小割。如果是树，只要找到从 S 到 T 的路径上的最小值就行了。本题是点-仙人掌，如果割掉的边不在环上，和树上的情况一样。如果割掉的边在环上，则要割掉两条边，其中有一条边一定是环上的最小边。可以把每个环上最小边去掉，再把这条边的权值加到环上其它边上，剩下的就是树上问题了。由于有修改操作，可以将每条边属于的环和每个环被去掉的边存下来，用 Link/cut tree 维护树结构。

40.3 时空复杂度

- 时间：$O(N \log N)$；
- 空间：$O(N)$。

41 JAN14.TAPAIR Counting The Important Pairs

41.1 题目大意

有一个无向图，询问去掉两条边后图不连通的方案数。

41.2 算法讨论

考虑图的一个生成树，如果有一个非树边 (u,v)，则我们称树上 u 到 v 的路径上所有点被这条非树边覆盖。如果去掉某个树边，必须再去掉所有覆盖它的非树边才会使图不连通。我们给每条非树边赋一个随机权值，每个树边的权值为所有覆盖它的非树边的权值异或和。可以发现，某个生成树的一组合法权值对于同一张图的任意生成树都是成立的。如果两条边权值相等，说明它们是某个生成树中的一条树边和唯一覆盖它的非树边，去掉这两条边后图不连通。如果一条边权值是 0，说明它是桥，去掉它和任意其它边后图不连通。为防止冲突，随机数的范围要大一些，最好是 64 位整数。

41.3 时空复杂度

- 时间：$O(N \log N)$；
- 空间：$O(N)$。

42 MAY12.TICKETS Selling Tickets

42.1 题目大意

有 N 道菜，M 个人，每个人只喜欢其中的两道菜，一道菜最多只能给一个人吃。询问最多能让几个人入场，使得无论是哪几个人，每人都能吃到一道喜欢的菜。$N \leq 200, M \leq 500$

42.2 算法讨论

求最大可行人数比较麻烦，可以改求最小不可行人数。相当于在一个无向图中选出一个最小的边数大于点数的子图，这样的图一定至少包含两个环，具体可能有三种形式：两个点间有三条不同路径，两个共用一个节点的简单环，两个由一条路径相连的简单环。第一种情况可以枚举起点三次 BFS 来寻找。后两种情况可以枚举起点 BFS 找两个环，再枚举连接点来寻找。

42.3 时空复杂度

- 时间：$O(N^2M)$；
- 空间：$O(N+M)$。

43 FEB15.CUSTPRIM Payton numbers

43.1 题目大意

定义三元组 (a,b,c)，$c \in \{24, 11\}$ 乘法 $(a_1,b_1,c_1)\cdot(a_2,b_2,c_2)$：

$$s = (a_1a_2 + b_1b_2 + c_1c_2) + (a_1b_2 + b_1a_2) + (c_1 + c_2)$$

$$t = \lfloor \frac{s}{2} \rfloor + 16(c_1 + c_2) - c_1c_2$$

$$A = (t - 2(a_1b_2 + b_1a_2) - (a_1c_2 + c_1a_2) + 33(a_1 + a_2) + (b_1b_2 - a_1a_2))$$

$$B = (t - 5(a_1b_2 + b_1a_2) - (c_1b_2 + b_1c_2) + 33(b_1 + b_2) + (2b_1b_2 + 4a_1a_2))$$

如果 s 是奇数，结果为 $(A-540, B-540, 24)$，否则为 $(A-533, B-533, 11)$。

定义单位元 A 是对于任何 B 都满足 $A \times B = B$ 的三元组，定义 zeroA 是对任何 B 都满足 $A \times B = A$ 的三元组，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。给定一个三元组, 判断它是不是素数。

43.2 算法讨论

令 ω 为方程 $\omega^2 = \omega - 3$ 的解，每个三元组 (a, b, c) 都有到域 $\mathbb{Z}[\omega] = \{a + b\omega | a, b \in \mathbb{Z}\}$ 的映射 $\phi(a, b, c) = (33 - 2a - c) + (b - a)\omega$。问题转化为判断域 $\mathbb{Z}$ 下的数 $a + b\omega$ 是否是素数。

定义共轭 $(a + b\omega)' = (a + b - b\omega)$，令 $Nx = xx'$，有以下结论：如果 x 不是整数，x 是素数当且仅当 Nx 是素数；如果 x 是整数，x 是素数当且仅当整数 x 是素数且 $|x| = 2$ 或 $|x| \neq 11$ 且 -11 是模 x 的二次非剩余。可以使用欧拉判别法和 Miller-Rabin 算法来解决这个问题。

43.3 时空复杂度

- 时间：$O(log|a|)$；
- 空间：$O(1)$。

44 JAN15.XRQRS Xor Queries

44.1 题目大意

有一个数列，初始为空。要求支持以下操作：在数列末尾添加一个数；删除数列末尾若干个数；查询区间第 k 大；查询区间小于等于某数的数的数量；查询区间中与某数异或和最大的数。

44.2 算法讨论

“查询与某数异或和最大的数”是 Trie 的经典应用。此处是在某个区间里查询。考虑可持久化，T_i 表示包含前 i 个数的 Trie，查询时在 T_R 里查，限制右端点，Trie 上的每个节点标记时间戳，查询时避免进入时间戳小于 L 的节点，限制左端点，实现区间查询。因为已经用了可持久化 Trie，区间 k 大相关问题可以通过在 Trie 上维护子树中数的个数来实现。

44.3 时空复杂度

- 时间：$O(N \log N)$；
- 空间：$O(N \log N)$。

45 DEC14.RIN Course Selection

45.1 题目大意

有 N 门课，M 个学期，每门课都必修，可以在任意一个学期学习。有些课会有前置课程，有 K 个前置关系。在第 j 学期学习第 i 门课的期望分数是 F_{ij}（在 0-100 范围内）。最大化期望平均分。$N, M, K \leq 100$

45.2 算法讨论

考虑最小割模型。先不考虑前置，对于每个课程建 M 个点，对于第 i 门课，第 $(j-1)$ 个点（$j=1$ 时为源点）连向第 j 个点的容量为 $100-F_{ij}$，第 M 个点连向汇的容量为 ∞。这个图的最小割就是最小损失分数。对于前置关系 $a \to b$，我们从课程 a 的第 i 个点（$i=0$ 为源）连一条容量为 ∞ 的边到课程 b 的第 $i+1$ 个点。

45.3 时空复杂度

- 时间：$O(maxflow(NM,(N+K)M))$；
- 空间：$O((N+K)M)$。

46 NOV14.SEAORD Sereja and Order

46.1 题目大意

有 N 个任务和两台电脑，第 i 个任务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。一台电脑同时只能运行一个任务。一个任务的两个子任务不能同时在两台电脑上运行。最小化用时，要输出方案。

46.2 算法讨论

答案的下界是 $\max(\sum A, \sum B, A_i+B_i)$，且最优解一定取到下界。如果最大值取到 A_i+B_i，由于 $\sum A \le A_i+B_i$，则 $\sum A - A_i \le B_i$，同理 $\sum B - B_i \le A_i$，所以安排好耗时最长的任务后，其他任务都可以填到两台机器的空闲时间里。

如果是其它情况，考虑每个任务，它的两个子任务可能都尽量早开始运行，或一个尽量早一个尽量晚，或都尽量晚，共四种情况。此时一定有一台机器持续不断运行，而另外一台机器可以做到仅有一段连续的空闲时间（即所有“尽量早”的任务之间，所有“尽量晚”的任务之间都不留空闲）。可以把所有任务随机打乱后贪心分配，但有可能分配到某个任务后四种情况都不行。幸运的是，合法最优解通常很多，所以只要多打乱尝试几次就行了。

46.3 时空复杂度

- 时间（期望）：$O(NC)$，其中 C 为期望尝试次数；
- 空间：$O(N)$。

47 JULY14.SEAEQ Sereja and Equality

47.1 题目大意

两个长度相同的数组相似当且仅当对于任意 i 满足 $CA(A_i) = CB(B_i)$，其中 $CX(x)$ 表示数组 X 中满足 $X_i < x$ 的个数。$F(A,B)$ 等于 $A[l..r]$ 与 $B[l..r]$ 相似且 $A[l..r]$ 逆序对不超过 E 个的数对 (l,r) 的数量。求对于所有排列，$\sum F(P_1,P_2)$。$T \le 10000, N \le 500, E \le 10^6$。

47.2 算法讨论

令 $F_{i,j}$ 表示 i 个数不超过 j 个逆序对的排列的个数，有递推式 $F_{i,j} = F_{i-1,j} - F_{i-1,j-i}$，则有结论：最终答案为 $\sum_{i=1}^{N}((N-i+1)*F_{i,E}*(\frac{N!}{i!})^2)$

47.3 时空复杂度

- 时间：预处理 $O(N^3)$，一个询问 $O(N)$；
- 空间：$O(N^3)$。

48 MAY14.SEINC Sereja and Subsegment Increasings

48.1 题目大意

有一个数组 A，每次可以选一个区间每个数加 1 模 4，求把它变成另一个数组 B 至少要操作几次。

48.2 算法讨论

两个数组太麻烦，考虑变成一个：$A_i \leftarrow (B_i - A_i) \mod 4$。问题变成，有一个数组 A，每次可以选一个区间每个数减 1，问至少要操作几次。直接操作的代价是 $\sum max(0, A_i - A_{i-1})$，但是我们可以选一些数加上 4 的若干倍来减少代价。具体方法是：设差分后的数组为 C，从左到右扫描，如果遇到 $C_i \in \{-1, 0, 1\}$，是不能优化的，直接计算进答案。遇到 $C_i \in \{-2, -3\}$，把它们的出现次数记录下来。遇到 $C_i \in \{2, 3\}$，此时可以优化，寻找一个没用过的 $C_j \in \{-2, -3\}, j < i$（优先用 -3），将 C_j 加 4，C_i 减 4（相当于原数组区间 $[j, i-1]$ 加 4），答案更优。最后 $\sum max(0, C_i)$ 就是答案。

48.3 时空复杂度

- 时间：$O(N)$；
- 空间：$O(N)$。

49 OCT12.MAXCIR Max Circumference

49.1 题目大意

给一个三角形 ABC 和 N 个操作，第 i 个操作使点 A 平移 (x_i, y_i)。最多使用这 N 个操作里的 K 个，求最大周长。

49.2 算法讨论

因为 $|BC|$ 不变，只要最大化 $|AC| + |AB|$。可以证明，能找到两个实数 u, v，最大化 $ux_a + vy_a$ 的同时最大化 $|AC| + |AB|$。因为所有达到最优解的点在一个焦点为 B, C 的椭圆上，只需求解 A 的切线方程就可以找到一组合适的 u, v，并找到对应的最大化方案。知道 u 和 v 后, 排序之后取前 K 个正权操作就能算

出 A 的坐标了。考虑 $ux_i + vy_i = ux_j + vy_j$ 的情况，可以解出两组 u, v，通过这些向量 (u, v) 可以确定 $ux_i + vy_i$ 和 $ux_j + vy_j$ 的大小。我们可以求出对于每一对操作时期相等的向量，按极角排序后取相邻向量夹角之间的任意向量作为 (u, v) 排序计算。更进一步，每次不需要重新排序，只要将涉及到的操作修改，再二分出 K 个正权操作就行了。另外 $ux_i + vy_i = 0$ 的情况也要考虑。这题还有一个问题是精度要求较高，不能直接开方，要先算出平方根的整数部分，再用一些小技巧算出更精确的平方根小数部分。

49.3 时空复杂度

- 时间：$O(N^2 \log N)$；
- 空间：$O(N^2)$。

50 SEPT13.TMP01 To Queue or not to Queue

50.1 题目大意

有 N 个操作，每个操作是在字符串结尾加一个字符或在开头删一个字符。维护字符串中本质不同的子串个数。$N \leq 1000000$，3 秒。

50.2 算法讨论

由于只要求子串种类数，不妨将字符串反转过来，操作就变成了开头加和结尾删。考虑只有开头加，可以先离线读入所有操作，构造后缀数组和 LCP 数组，真正做的时候每在开头加一个字符，就把对应的后缀加入线段树，就实现了一个“离线动态后缀数组”，很容易统计答案。考虑删除操作，相当于每个后缀长度减 1，但是如果减到了某两个后缀的 LCP 就会出问题。为了解决这个问题，我们规定，线段树中不得出现某个后缀（指删过字符的后缀）是另一个后缀的前缀，如果出现了，就要把是前缀的后缀删除。这样做不影响答案，同时使删除操作只影响后缀长度，不影响线段树中后缀的 LCP，避免了问题。需要注意的是，此题使用这种算法，对常数的要求很高。

50.3 时空复杂度

- 时间：$O(N \log N)$；
- 空间：$O(N \log N)$ （使用了 $O(N \log N)/O(1)$ 的 RMQ）。

51 SEPT13.TWOROADS Two Roads

51.1 题目大意

平面上有 N 个点，要求作两条直线，使每个点到直线的最短距离平方和最小。不存在重复点和三点共线。

51.2 算法讨论

两条直线所成角的角平分线（在两条相互垂直的直线上）将平面分成 4 个部分，每个部分都由某条固定的直线控制。确定了角平分线后就可以用线性回归计算最优的直线位置。可以发现，对于每种控制情况，总是可以找到一组角平分线，一条过两个点，另一条过一个点。可以枚举两个点，再按顺序枚举垂足，维护每个点被哪条直线控制，计算并更新答案。

51.3 时空复杂度

- 时间：$O(N^3 \log N)$；
- 空间：$O(N)$。

52 JUNE13.SPMATRIX Count Special Matrices

52.1 题目大意

一个 $N \times N$ 整数矩阵是特殊的，当且仅当：

- $A_{i,i}$=0
- $A_{j,i} = A_{i,j} \in [1, N-2]$
- $A_{i,j} \leq \max(A_{i,k}, A_{k,j})(i, j, k \in [1, N])$
- 对于所有 $k \in [1, N-2]$，至少存在一个 $A_{i,j} = k$

给出 N，询问矩阵个数模 10^9+7。$N \leq 10^7$，100000 个询问。

52.2 算法讨论

答案等于 $\frac{n!(n-1)!}{3\times 2^{n-1}}(\frac{3n}{2} - 2 - \sum\limits_{i=1}^{n-1}\frac{1}{i})$。可以先算出所有答案再查表回答询问。需要线性筛求逆元。注意乘法和取模的常数。

52.3 时空复杂度

- 时间：预处理 $O(N)$，一组询问 $O(1)$；
- 空间：$O(N)$。

53 JAN13.CUCUMBER Cucumber Boy and Cucumber Girl

53.1 题目大意

有 B 个 $N \times N$ 矩阵，数对 (a, b) 表示一个新的矩阵 B，满足 $B_{i,j} = \sum\limits_{k=1}^{N} A_{a,i,k} \times A_{b,j,k}$。一个排列 P 是好排列当且仅当存在 i 使 B_{i,P_i} 为奇数。数对 (a, b) 是好的当且仅当它对应的矩阵 B 有奇数个好排列。询问有多少好数对。

- $N \leq 60$
- $B \leq 8000$

53.2 算法讨论

把矩阵 B 变成 $B_{i,j} \leftarrow (B_{i,j}+1) \mod 2$。此时好数对对应的矩阵 B 的行列式为奇数。此时 $B_{i,j} = \sum_{k=1}^{N}(A_{a,i,k} \times A_{b,j,k}+1) \mod 2$，可以在每个矩阵后加上全是 1 的一列，此时 $B = A_a \times A_b^T$（在模 2 意义下）。根据 Binet-Cauchy 定理，设 A_{ij} 为矩阵 A_i 删去第 j 列，则 $|B| = \sum_{k=1}^{N}|A_{ai}|+|A_{bi}|$。对 A_i 消元后做一些处理就能求出所有 $|A_{ij}|$ 了，可以压位优化。

53.3 时空复杂度

- 时间：$O(N^2B+B^2)$；
- 空间：$O(N^2B)$。

54 AUG15.DISTNUM Simple Queries

54.1 题目大意

有一个数列 A_i，要求支持以下 5 种操作

- 令 S 为某区间内不同的元素构成的有序集合。你需要求出

$$\sum_{1\leq i<j<k\leq |S|} S_iS_jS_k \pmod{10^9+7}$$

- 插入一个数
- 删除一个数
- 修改一个数
- 询问某区间内不同的数的个数

54.2 算法讨论

本题有插入和删除，可以通过离线把它们都变成修改。操作 1 比较复杂，可以用容斥法考虑，假设区间内的数各不相同，设它们的一次方，二次方，三次方之和分别为 S_1, S_2, S_3，则答案为 $\frac{S_1^3-3S_1S_2+2S_3}{6}$。再考虑操作 1 和操作 5 中重复的元素的问题。我们对每个位置的数维护一个“相同的数上一次出现位置” Pre_i（第一次出现则为 0），每个数原来坐标为 i，现在变为 (i, Pre_i)，原来的一维区间查询 $[l,r]$ 变为二维矩形查询 $[l,r][0,l-1]$，可以用二维数据结构维护。

54.3 时空复杂度

- 时间：$O(N\log^2 N)$；
- 空间：$O(N\log^2 N)$。

55 SEPT14.FIBTREE Fibonacci Numbers on Tree

55.1 题目大意

给定一棵点带权的树（初始全为 0），支持 4 种操作：给一条路径上的点权加上一个 Fibonacci 数列，询问子树权值和，询问路径权值和，回退到第 i 次操作后。所有数值模 10^9+9。强制在线。

55.2 算法讨论

我们可以找到一个整数 x 满足

$$x^2 \equiv 5 \pmod{10^9+9}$$

，从而

$$x \equiv \sqrt{5} \pmod{10^9+9}$$

，然后使用 Fibonacci 数列的通项公式。Fibonacci 数列是两个等比数列相加得到的，可以分别考虑两个等比数列。我们需要实现一个“区间加等比数列（公比始终不变）”的操作。由于公比不变，每个区间只要维护首项就行了，这样的标记是可以合并的。线段树上每个节点维护两个公比的正着、倒着共计 4 个首项。此处的标记是永久标记，不下传。至于树上问题和可持久化，只要套用树链剖分和可持久化线段树就可以实现。

55.3 时空复杂度

- 时间：$O(N \log^3 N)$；
- 空间：$O(N \log^2 N)$。

56 JUNE14.SEAARC Sereja and Arcs

56.1 题目大意

数轴上有 N 个点 $(i,0), 1 \le i \le N$，每个点一个颜色，相同颜色点之间连接圆弧，求相交的异色圆弧对数。

56.2 算法讨论

相交的异色圆弧对数不好做，可以改为求 圆弧对数 − 同色圆弧对数 − 相离的异色圆弧对数 − 相互包含的异色圆弧对数。前三者可以 $O(1)$ 或 $O(N)$ 计算。考虑最复杂的“相互包含的异色圆弧对数”。我们称点数超过一个阈值 B 的颜色称为“大”的颜色（显然这种颜色数量不超过 $\frac{N}{B}$）。把相互包含的异色圆弧对分成三类：外侧为“大”颜色，内侧为“大”颜色且不属于第一类，不属于前两类。对于前两者，可以枚举“大”颜色，对于每种情况线性计算。最后一种情况，可以使用复杂度与颜色点数相关的方法统计，需要用到数据结构。

56.3 时空复杂度

- 时间：$O(\frac{N^2}{B} + BN \log N)$；
- 空间：$O(N)$。

57 APR14.GERALD08 Chef and Tree Game

57.1 题目大意

两个人在一棵树上玩游戏，树上有两种颜色的边，每个人操作时可以删除一条自己颜色的边，把与根节点不连通的部分删除。两人轮流操作，谁先不能操作谁输。问两个人分别先手时谁必胜。

57.2 算法讨论

本题中的游戏是经典的多人博弈游戏 Hackenbush，对于此类游戏学术界目前有很多研究。本题是树上二人的特殊情况。我们给每个局面定义一个权值 F，$F > 0$ 第一人必胜，$F < 0$ 第二人必胜，$F = 0$ 后手必胜。则有结论：与根节点用第一类边相连的权值为 x 的点，对根的贡献为 $\frac{x+p}{2^{p-1}}$（p 为最小的满足 $x + p > 1$ 的正整数）；如果连接的是第二类边，对根的贡献为 $\frac{x-p}{2^{p-1}}$（p 为最小的满足 $x - p < -1$ 的正整数）。注意此处的权值不能用浮点数（精度不够），由于分母一定是 2 的幂，可以使用平衡树启发式合并来实现高精度运算。

57.3 时空复杂度

- 时间：$O(N \log^2 N)$；
- 空间：$O(N \log N)$。

58 MARCH14.STREETTA The Street

58.1 题目大意

有 N 个商店，买一个东西要付的钱是价格 + 税费，初始每个商店都没有商品，税费为 0，要求实现以下三种操作：给某区间内的所有商店里加一个商品（价格为等差数列），给某区间内的所有商店的税费都增加一定值（等差数列），询问取某个商店买一件东西最多要花多少钱。M 次操作。$N \leq 10^9, M \leq 300000$

58.2 算法讨论

商店编号可以离散化。本题要维护两个数列，一个需要区间加等差数列，另一个需要区间与等差序列取 max。区间加等差序列与本次作业中另外一题（题号 QUERY）相同，此处不介绍。区间与等差序列取 max 时，考虑两个标记的合并，由于两个等差数列（一次函数）取 max 最多只有一个交点，合并标记时如果有相交，把一个标记放在当前节点上，另一个标记向产生交点的那一侧下传。只需要往线段树的一个儿子上继续递归。

58.3 时空复杂度

- 时间：$O(M \log^2 M)$；
- 空间：$O(M)$。

59 FEB14.COT5 Count on a Treap

59.1 题目大意

有一个大根 Treap，支持三种操作：插入一个给定优先级和权值的点，删除一个点，询问两点间距离。

59.2 算法讨论

只要能求深度和 LCA，就能求距离。求 LCA 非常简单，就是两个节点对应区间中优先级最大的点。考虑互为祖先-子孙的两个节点，对应的区间上不存在优先级更大的节点。所以一个节点的深度就是它向左和向右的上升序列的长度和。由于两侧的情况相似，仅考虑一侧，可以在线段树的每个节点上维护最大优先级和上升序列的长度，查询和更新时需要用到“查询节点对应的区间中首项大于 k 的上升序列长度”操作，此操作需要向一侧子树递归。

59.3 时空复杂度

- 时间：$O(N\log^2 N)$；
- 空间：$O(N)$。

60 DEC13.REALSET Petya and Sequence

60.1 题目大意

给定一个 N 个数的序列 A，求是否存在非全 0 的序列 B，满足对于任意 j，

$$\sum_{i=0}^{N-1} A_i B_{(i+j) \mod N} = 0$$

，$N \le 30000, |A_i| \le 1000$，多组询问，$\sum N \le 150000$。

60.2 算法讨论

齐次线性方程组若有唯一解，一定为零解。本题实际上要判断一个循环矩阵是否满秩，即判断行列式是否为 0。对于循环矩阵，构造多项式 $f(x) = \sum\limits_{i=0}^{N-1} A_i x^i$，则行列式值为 $\prod\limits_{i=0}^{N-1} f(\omega_N^i)$，其中 ω_N 为 N 次单位根。此处涉及一个任意点数的 FFT，可以使用 Bluestein's FFT Algorithm 实现（此算法的思想是构造两个向量，使得它们的卷积再乘一组系数就是 DFT 的结果，将任意点数 FFT（对因数无要求）转化为卷积问题，使用 2^K 点数的 FFT 或分治乘法解决）。注意到 N 个复数相乘会爆精度，可以改为在模意义下计算，要取 2 个 50000 以上的 $kN+1$ 形式的质数作为卷积的模，再取 2 至 3 个 10^9 以上的 $c \times 2^K + 1$ 形式的质数作为 NTT 的模，实现较为繁琐。

60.3 时空复杂度

- 时间：找适合条件的模数直接暴力，由于满足条件的质数很多，所以很快。其它部分复杂度 $O(N \log N)$；
- 空间：$O(N)$。

61 NOV13.QPOINT Queries With Points

61.1 题目大意

有一些无交的简单多边形（不一定凸）的区域。每次询问一个点在哪个简单多边形区域内。强制在线。

61.2 算法讨论

考虑扫描线。以每个顶点的 x 坐标来划分区间（有些线段会被切开），每条线段的出现时间是连续的一些区间，且对于某个线段集合，只要这些线段都存在，线段上的点的相对顺序是不变的。所以可以对所有线段的端点排序，从左到右扫描，用平衡树来维护线段顺序。因为强制在线，要用可持久化平衡树，每个区间保留一个状态。查询时，只要找到对应区间的平衡树，查出这个点下方的线段条数就可以知道内外，如果在多边形内，再查出下方最近的线段所属区域编号。与 y 轴平行的线段要特判。

61.3 时空复杂度

- 时间：$O((N+Q)\log N)$；
- 空间：$O(N)$。

62 AUG13.PRIMEDST Prime Distance On Tree

62.1 题目大意

求一棵树上任取两点，路径长度是质数的概率。$N \le 50000$

62.2 算法讨论

我们可以求出树上每种不同长度的路径条数。可以使用点分治 + 卷积来实现。

62.3 时空复杂度

- 时间：$O(N \log^2 N)$；
- 空间：$O(N \log N)$。

63 APR13.STRQUERY String Query

63.1 题目大意

动态维护字符串，允许在开头、正中间、结尾插入删除字符和询问某个子串的出现次数。涉及的字符串总长度不超过 1500000。

63.2 算法讨论

只有开头插入删除，可以用后缀平衡树。考虑有结尾插入删除的情况，我们用两个后缀平衡树分别维护左半个字符串和右半个字符串，如果某一个变成空，就把它们暴力平分（复杂度均摊后一样），查询时中间的那些可以暴力 KMP，这样就形成了一个“字符串 Deque”。再考虑有中间插入删除的情况，可以维护两个“字符串 Deque”，一个维护严格的左半边，另一个维护严格的右半边。

63.3 时空复杂度

- 时间：$O(|S|\log|S|)$；
- 空间：$O(|S|)$。

64 FEB13.QUERY Observing the Tree

64.1 题目大意

给定一棵初始点权全 0 的树，支持三种操作：树上一个路径上的点权加上一个等差数列，求某个路径点权和，回退到第 i 次操作之后。

64.2 算法讨论

首项 A，公差 B 与首项 C，公差 D 的两个等差数列相加可得到一个首项 $(A+B)$，公差 $(C+D)$ 的等差数列。所以“区间加等差数列”可以在线段树的节点上打标记 (首项, 公差) 来实现，标记可以轻松地合并。这里的标记是永久标记，不下传。树上操作和回退可以直接套用树链剖分和可持久化线段树。

64.3 时空复杂度

- 时间：$O(N\log^2 N)$；
- 空间：$O(N\log^2 N)$。

65 NOV12.MARTARTS Martial Arts

65.1 题目大意

两个队伍各有 N 人参加拳击比赛，共比 N 场，每个选手一场，主队第 i 人和客队第 j 人比赛，两队分别得分 $A_{i,j}, B_{i,j}$，设两队得分为 H 和 G，则你要安排比赛最大化 $make\_pair(H-G,H)$，但是你安排好后，对方会在最大化 $make\_pair(G-H,G)$ 的前提下故意取消一场比赛。求最优答案。

65.2 算法讨论

可以将所有边排序，依次加入图中，求必须匹配最后一条边的情况下的带权最大二分图匹配。动态匹配可以通过将 KM 算法经过少量修改实现。

65.3 时空复杂度

- 时间：$O(N^4)$；
- 空间：$O(N^2)$。

66 SEPT12.PARADE Annual Parade

66.1 题目大意

有 N（$N \leq 250$）座城市 M 条道路，可以安排若干条游行路线（每条路线必须至少经过一条边，不能单点），费用为经过的道路长度之和（可以重复经过，费用累计）+ 起点不等于终点的游行路线数 $\times C+$ 没有路线经过的城市数 $\times C$。给定若干个关于 C 的询问，回答最小费用。

66.2 算法讨论

这个问题的难点有两个：边允许重复经过，点允许不经过。对于边的问题，可以把原图补成完全图，其中边 (i,j) 的长度为原图中 i 到 j 的最短路，此时原图中一组允许相交的路径对应完全图中一组不允许相交的路径。对于点的问题，我们修改一下题目条件，允许单点路径（但不算“起点等于终点”，需要额外费用）而不允许有点不经过，效果是一样的。此时原题变为一个简单的点覆盖问题，可以把每个点拆成“入点”和“出点”，用匹配来解决。如果有一个匹配数为 K，费用为 S 的匹配，说明有 $(N-K)$ 条路径未闭合或是单点，总费用为 $S+(N-K)\times C$。对于多个询问，预处理时每次增广 1 流量，记下每次的费用增量，回答询问时二分最优位置（后面的费用全改成 C）。

66.3 时空复杂度

- 时间：$O(N^3+N\times SPFA(N,N^2)+Q\log N)$；
- 空间：$O(N^2)$。

67 AUG12.GTHRONES A Game of Thrones

67.1 题目大意

纸上有 N（$N \leq 500$）个不同的数，每种数有若干个，先手可以先选一个当前数字，然后轮流操作：擦去当前数字，选择另一个数作为新的当前数字，两个数之间必须只相差一个质因子。不能操作者输。问是否先手必胜，选哪些数作为初始当前数字可以必胜。

67.2 算法讨论

由于一个数只可能有奇数个或偶数个质因子，连边后一定是二分图。原题变为经典的二分图博弈问题，所有在某个最大匹配中未匹配的点都是可选的必胜点。如果只求是否必胜，一次最大流就能解决。如果求哪些点可以是未匹配点，需要依次尝试，可以使用“退流”法。

67.3 时空复杂度

- 时间：$O(maxflow(N,N^2)+N\times back(N,N^2))$，$back(N,M)$ 为退掉一条边的流量再重新求最大流的复杂度；
- 空间：$O(N^3)$。

68 JULY12.DGCD Dynamic GCD

68.1 题目大意

有一棵 N 个节点的树，两种操作：树上一条路径每个点点权加上某个值，求树上一条路径里所有点的 GCD。

68.2 算法讨论

GCD 有这样的性质：$GCD(A,B,C,\cdots)=GCD(A,B-A,C-B,\cdots)$，可以对所有点差分，区间修改变成单点修改，区间 GCD 变成“区间 GCD，前缀和”的 GCD，线段树可实现。树上问题直接套用树链剖分。

68.3 时空复杂度

- 时间：$O(N\log^2 N\log A)$；
- 空间：$O(N)$。

69 JUNE12.COOLNUM Cool Numbers

69.1 题目大意

一个各位数之和为 K 的数，如果取其中 1 到 3 位（和为 S），$(K-S)^S$ 是原数的倍数，则它是 Cool Number。最多 100000 组询问，每次一个不超过 1000 位的数，询问小于等于它的最大 Cool Number $LC(N)$ 和大于它的最小 Cool Number $RC(N)$。

69.2 算法讨论

显然，非零数位不超过三个的数都是 Cool Number，一个 L 位数 N，可以在 $O(L)$ 的时间内求出这类数中的 $LC(N),RC(N)$。对于不平凡的 Cool Number，因为 $(9L-27)^{27}>10^{L-1}$，L 不超过 80，可以枚举 $(K-S)$ 再枚举 $(K-S)^{27}$ 的所有范围内的因数来找出所有这样的 Cool Number，总数不超过 40000 个，回答询问时二分。

69.3 时空复杂度

- 时间：$O(P + T\log N\log Cnt)$，P 为爆搜预处理的时间,Cnt 为不平凡的 Cool Number 的个数；
- 空间：$O(cnt)$。

70 JUNE12.MATCH Expected Maximum Matching

70.1 题目大意

有一个左边 N 个点 $(N \leq 5)$ 右边 M 个点 $(M \leq 100)$ 的二分图，每条边有一个出现概率，问期望最大匹配。

70.2 算法讨论

考虑 Hall 定理，如果有一个匹配左边的点集是 U，则对于每个子集 $S \subseteq U$，相邻的右边点数不少于 $|S|$。我们称这样的集合 S 为 Hall 点集，这样的点集的集合 $U = \{S_1, S_2, \cdots\}$ 称为 Hall 点集集合。考虑 DP，$F_{U,i}$ 表示仅考虑右边前 i 个点，点集集合 U 是 Hall 点集集合的期望。点集集合最多有 2^{2^n} 个，但是很多都不可能满足条件，合法的只有 4000 个左右，可以 BFS 得到。

70.3 时空复杂度

- 时间：$O(CM)$，C 为合法点集集合数量；
- 空间：$O(CM)$。

71 APRIL12.CONNECT Find a special connected block

71.1 题目大意

有一个 $N \times M(N, M \leq 15)$ 的矩阵，每个格子填一个 $[-1, N \times M]$ 的整数，要求选一个连通块，不包含-1 格子且包含至少 K 种正数 $(K \leq 7)$，每个格子有代价，求最小代价。

71.2 算法讨论

如果权值不超过 K，相当于斯坦纳树问题，可以通过枚举子集 + 最短路来实现。权值比较大的情况，可以随机将权值映射到 $[-1, K]$ 的范围内，这样的结果一定不比原问题更优，随机若干次就有很大概率得到最优解。

71.3 时空复杂度

- 时间：$O(SPFA(2^k NM, 3^k NM))$；
- 空间：$O(2^k NM)$。

72 MARCH12.EVILBOOK Evil Book

72.1 题目大意

有 N($N \le 10$) 个敌人，打倒第 i 个敌人代价为 C_i，可以得到 M_i 点魔力，只有打倒第一次才有魔力奖励。可以消耗 X 点魔力寻求帮助（必须手里有至少 X 点）将某个敌人的 C_i 和 M_i 都除以 3。询问至少付出多少点代价收集到 666 点魔力。

72.2 算法讨论

设 k_i 为降第 i 个人属性的次数。因为付出代价没有上限，而降属性有上限（必须有 X 点魔力），可以限制按照 k_i 递增的顺序打倒敌人。由于我们要花最小代价收集 666 点魔力，有些 k_i 是肯定不优的，可以证明，每个 k_i 最多只有 4 种有效取值。于是可以搜索 + 剪枝。

72.3 时空复杂度

- 时间：$O(4^N)$；
- 空间：$O(N)$。

73 MARCH12.CIELQUAK Ciel and Earthquake

73.1 题目大意

有 $R \times C$ 的网格状道路，每条路的连通概率都为 P，问路口 $(1,1)$ 和 (R,C) 连通概率。$R \le 8, C \le 10^{18}$。

73.2 算法讨论

考虑 C 比较小的情况，可以使用插头 DP 计算概率。打表观察可以发现，随着 C 增大，$\frac{Ans_{R,C+1}}{Ans_{R,C}}$ 的比值趋向一个常数。可以取一个 M，计算 $Ans_{R,M} \times (\frac{Ans_{R,M+1}}{Ans_{R,M}})^{(C-M)}$。M 取 40 左右精度就达到要求了。有效的连通性对应的转移可以 BFS 预处理出，每个点状态数只有 3000 多个。

73.3 时空复杂度

- 时间：$O(RMCnt)$，Cnt 为有效连通性状态数；
- 空间：$O(RMCnt)$。

74 FEB12.FINDSEQ Find a Subsequence

74.1 题目大意

给一个 1 到 5 的排列 B 和数组 A，找出 A 的一个 5 个数的子序列，要求相对顺序和 B 一样。$N \le 600$

74.2 算法讨论

枚举第二个数和第四个数，然后可以贪心地选第一个数和第五个数（如果它比第三个数大，取所有合法数中最大的，否则取最小的），此时合法的第三个数大小在某个区间内，可以判断有没有这个数，然后暴力找到一组可行解。需要预处理前缀（后缀）比某个数大（小）的数以及一定范围内值在一定范围内的数的个数，可以 $O(N^2)$ 预处理。

74.3 时空复杂度

- 时间：$O(N^2)$；
- 空间：$O(N^2)$。

75 FEB12.FLYDIST Flight Distance

75.1 题目大意

有一个 N 个点的带权无向图，可以每次花一点代价将一条边的长度 +1 或-1。问使得对于所有边 (i, j)，i 到 j 的最短路不小于边 (i, j) 的长度的最小代价。$N \leq 10$。

75.2 算法讨论

题目的条件等价于对于所有的简单环，环上每一条边长度都大于等于其它边的和。只要考虑所有无弦环，其它所有环会自动满足要求。本题变为一个线性规划问题，可以使用单纯形算法解决。

75.3 时空复杂度

- 时间：单纯形算法理论最坏时间复杂度是指数级的，但在本题上效果很好；
- 空间：$O(NMC)$，C 为无弦环个数。

76 JAN12.CARDSHUF Card Shuffle

76.1 题目大意

有 N 张牌，重复 M 次以下洗牌动作：拿走 A 张牌，再拿走 B 张牌，将 A 张牌直接放回牌堆上方，再拿走 C 张牌，将 B 张牌一张一张（倒序）放回牌堆上方，最后将 C 张牌直接放回牌堆上方。

76.2 算法讨论

直接用平衡树模拟所有操作。

76.3 时空复杂度

- 时间：$O(M \log N)$；
- 空间：$O(N)$。

77 JAN12.MISINT2 Misinterpretation 2

77.1 题目大意

给定 L,R，问把偶数位置的字符拿到开头后不变的长度在 L 到 R 之间的小写字母字符串个数。

77.2 算法讨论

每个重排相当于一个置换，如果循环数是 $f(n)$，方案数就是 $26^{f(n)}$。当 n 为奇数时，最后一个字符不变，所以 $f(2n+1)=f(2n)+1$。现在只需要考虑偶数情况。T 组询问。$T \le 5, L, R \le 10^{10}, R-L \le 50000$

令 $ord(x)$ 为 2 模 x 的阶，有结论 $f(n) = \sum\limits_{p|n+1, p>1} \frac{\phi(p)}{ord(p)}$。我们需要对 $n \in [L-1, R]$ 计算 $f(n)$，因为要枚举所有因数，可以先对 $[L-1, R+1]$ 分解质因数，用类似筛法的方法一起分解。再考虑怎么求 $ord(p)$，可以发现当 ab 互质时，$ord(ab) = lcm(ord(a), ord(b))$，所以只要对质数的幂计算 $ord(p)$ 就行了，可以对小于 100000 的数预处理出来，剩下 $[L-1, R+1]$ 里每个数只要额外算一个质数的 $ord(p)$。

77.3 时空复杂度

- 时间：$O(\frac{R^{\frac{3}{4}}}{\log R} + R^{1.5} + T(R-L)(C+\sqrt{R}))$，C 为因数个数；
- 空间：$O(\sqrt{R} + (R-L)\log R)$：。

78 DEC11.SHORT2 Short II

78.1 题目大意

给定一个质数 p，问有多少对 $a, b > p$ 满足 $(a-p)(b-p)|ab$，T 组数据

- $T \le 5$
- $p \le 10^12$

78.2 算法讨论

简化题意，将 a 和 b 都加 p，题目变成有多少对正整数 a,b 满足 $ab|(a+p)(b+p)$，等价于 $ab|p(a+b+p)$，设 $kab = p(a+b+p)$，分三种情况。

当 $p|a$ 且 $p|b$ 时，设 $a = xp, b = yp$，则 $kxy = x+y+1$，可以发现 (x, y) 有且仅有 5 种解:$(1,1),(1,2),(2,1),(2,3),(3,2)$。

当 $p \nmid a$ 且 $p \nmid b$ 时，若 $a = b$，只有 $a = b = 1$ 满足要求。若 $a \ne b$，不妨设 $a < b$，令 $k = pt$，则 $tab = a+b+p$，$b = \frac{a+p}{at-1}$。且 $a+b+p \ge ab$，则 $p+1 \ge (a-1)(b-1)$，a 有上界 $1+\sqrt{p+1}$。设 $d = at-1$，此时对于合法的 (a, b) 满足：

- $p \nmid a$
- $d|(a+p)$
- $a|(d+1)$

- $b = \frac{a+p}{d} > a$

此时 b, d 至少有一个不超过 $\sqrt{p+1+\sqrt{p+1}}$，可以在 $O(\sqrt{p})$ 的时间内枚举 b 算出对应的 a, d 以及枚举 d 算出对应的 a, b 来找出所有 (a, b)

当 $p|a$ 和 $p|b$ 有一者满足时，答案为前一种情况的两倍。因为前一种情况中一组可行解 (a, b) 对应这种情况的两组可行解 $(a, \frac{p(a+p)}{b})(\frac{p(b+p)}{a}, b)$

78.3 时空复杂度

- 时间：$O(T\sqrt{p})$；
- 空间：$O(1)$。

79 NOV11.LUCKYDAY Luckdays

79.1 题目大意

有一个数列 $S_1 = A, S_2 = B, S_i = (XS_{i-1} + YS_{i-2} + Z) \mod P (i > 2)$，Q 个询问，求有多少个 $k(L_i \le k \le R_i)$ 满足 $S_k = C$。

- P 是质数，$P \le 10007$
- $Q \le 20000$
- $L_i, R_i \le 10^{18}$

79.2 算法讨论

X, Y 有一个为 0 时循环节长度是 $O(P)$，可以暴力。X, Y 均不为 0 时循环节长度可能达到 $O(P^2)$，可以使用矩阵乘法（实际使用时不一定要写矩阵，只要知道有结合性就行了）+ 大步小步来解决。

79.3 时空复杂度

- 时间：$O(P^{1.5} + Q \log P)$；
- 空间：$O(P^{1.5})$。

80 OCT11.BAKE The Baking Business

80.1 题目大意

有 Q 个事件。第一种：

```
I 产品编号[.大小编号] 省编号[.城市编号[.地区编号]] 性别 年龄 出售数量
```

表示出售的一种产品的细节，方括号部分可能缺失，表示这个信息丢失（不统计进相应类）。

第二种：

```
Q 产品编号[.大小编号] 省编号[.城市编号[.地区编号]] 性别 起始年龄[-结束年龄]
```

表示一个询问，方括号部分缺失或者产品编号、省编号为 -1 表示不限定这个条件。要求输出满足条件的商品出售总数。

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。$S \le 100000$

80.2 算法讨论

开一个七维数组直接统计即可。

80.3 时空复杂度

- 时间：$O(Q)$；
- 空间：$O(1)$；
- 常数较大。

81 SEPT11.SHORT Short

81.1 题目大意

给两个数 n,k，找出所有数对 $(a,b)(n \le a, b \le k)$，并且 $(a-n)(b-n)|(ab-n)$，输出满足条件的数对数目。

- $n \le 100000$
- $k \le 10^{18}$

81.2 算法讨论

$n = 0$ 时答案为 $(k-n-1)^2$，需要高精度。考虑 $n > 0$ 的情况，设 $p(a-n)(b-n) = ab-n$，假设 $a \le b$，$b = n + \frac{n(a-1)}{p(a-n)-a}$，a 有上界 $2n + \sqrt{2n^2-n}$，设 $d = p(a-n) - a$，$b = n + \frac{n(a-1)}{d}$，$p = \frac{d+a}{a-n}$，此时可以枚举 a, 再枚举 d（即枚举 $n(a-1)$ 的因数）算出 b 判断是否合法。当 a 很大时（本题可以取 $a \ge n + 3000$）枚举 d 比较困难，改为枚举 p。

81.3 时空复杂度

- 时间：$O(nC)$，C 为对于每个 a 的枚举次数，本题不超过 1200；
- 空间：$O(n \log n)$。

82 SEPT11.CNTHEX Counting Hexagons

82.1 题目大意

有无限根木棍，长度在区间 $[1,n]$ 中。要求用这些木棍拼六边形，满足：最长木棍长度大于 L，其它木棍长度小于 X，同样长的木棍不超过 K 根。计算有多少合法六边形。

- $N \le 10^9$
- $N - L \le 100$

82.2 算法讨论

先枚举最长木棍长度，然后考虑数位 dp，$f[i][j][k][l]$ 表示考虑到第 i 位，进位为 j，5 根木棍长度和是否大于最长木棍（1bit），6 根木棍从大到小的相等关系（5bit）的方案数。为了减少转移的复杂度，数位 dp 应使用二进制（此时进位可能跨多位，但绝对值不超过 5）。为了防止长度为 0 出问题，可以将所有木棍长度 -1。

82.3 时空复杂度

- 时间：$O(Base^5 MaxCarry^2 (N-L) \log_{Base} N)$，Base 为选用的进制（建议用二进制），MaxCarry 为最大进位；
- 空间：$O(MaxCarry \cdot \log_{Base} N)$；
- 有大常数 2^6。

83 AUG11.SHORTCIR Shortest Circuit Evaluation

83.1 题目大意

给一个含 and,or,not, 变量和括号的短路逻辑表达式和每个变量为 true 的概率，允许在不改变结果的情况下交换表达式中变量顺序，求最小期望计算次数。表达式中所有涉及优先级的地方都加了括号。

- 表达式长度 $|S| \leq 30000$
- 变量数目 $N \leq 1000$
- 数据组数 $T \leq 50$

83.2 算法讨论

先建表达式树，发现只能调整某个节点的孩子的顺序。设某棵子树期望计算次数为 F_i，为 true 的概率为 G_i，则将孩子按照 $\frac{F_i}{G_i}(or)$ 或 $\frac{F_i}{1-G_i}(and)$ 从小到大排序，结果最优。

83.3 时空复杂度

- 时间：$O(T|S|\log|S|)$；
- 空间：$O(|S|)$。

84 JUNE11.CLONES Attack of the Clones

84.1 题目大意

定义布尔函数 $f(A)=B$，A 为 n 个二进制位，B 为 0 或 1。有下面 4 类特殊函数，它们的集合分别用 Z, P, D, A 表示：

- Z 是 0-保留值函数集合，满足 $f(000\cdots0)=0$

- P 是 1-保留值函数集合，满足 $f(111\cdots 1)=1$
- D 是自对偶函数集合，满足 $\neg f(A_1A_2\cdots A_n)=f(\neg A_1\neg A_2\cdots\neg A_n)$
- A 为仿射函数集合，满足如果 $f(A_1\cdots C\cdots A_n)=f(A_1\cdots D\cdots A_n)$，则上式对于任意 $A_i(i\in[1,n]\text{且不在}C,D\text{的位置上})$ 都成立，其中 C 和 D 为某个位置 i 上的数。

Q 个询问，每个询问给定一个上述集合的组合（一个含有并、交、差、补和括号的表达式），要求输出对应的集合里有几个函数，模 1000003。$n,Q,|S|\le 100$

84.2 算法讨论

函数是否属于这 4 个集合有 16 种情况，可以先打表找规律推公式算出“属于某些集合的函数个数”，再容斥算出“仅属于某些集合的函数个数”，然后就是简单的表达式计算了。

84.3 时空复杂度

- 时间：$O(\log\phi+Q|S|)$；
- 空间：$O(|S|)$。

85 DEC11.HYPER Hypertrees

85.1 题目大意

3-超图类似普通的图，但每条边连接 3 个点。3-超树是去掉任意一条边都不连通的连通 3-超图。给定 N，询问有 N 个点的带标号的本质不同的 3-超树有几个。$N\le 17$

85.2 算法讨论

考虑一个点数大于 3 的点双连通超树，每条边连接的 3 个点中，有且仅有 1 个点是叶子（如果为 0 个叶子，则它不是超树，去掉这条边后仍然连通；如果为 2 个叶子，则它一定不是双连通的，3 个叶子仅出现于 3 个点的情况），去掉这些叶子后变为一个普通的点双连通图，可以暴搜这种图的数量，然后暴搜用这些双连通分量组装成超树的个数。可能要运行很长时间，但 $N\le 17$ 可以打表。

85.3 时空复杂度

本题可以打表

86 NOV11.DOMNOCUT Colored Domino Tilings and Cuts

86.1 题目大意

一个 N 行 M 列的矩形棋盘。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。一个格子与另

一个格子相邻当且仅当他们有公共边。每个字母对应一种颜色。棋盘的割是指一条竖直或水平的直线将棋盘分成两半，而且这两半都是合法的棋盘覆盖染色。给你 N 和 M。你要构造一个棋盘覆盖染色使得，棋盘的割的数量最少。如果有多个解，你要使得使用的颜色最少。T 组数据。

- $T \le 3000$
- $N, M \le 500$

86.2 算法讨论

无解当且仅当 $N \times M$ 为奇数。当 N,M 足够大时一定有无割的 3 种颜色的方案，因为可以构造 5×6 和 6×8 的情况，并扩展到 $(5+2k) \times (6+2t)$ 和 $(6+2k) \times (8+2t)$ 的情况。其它分 $1 \times M, 2 \times 2K, 2 \times (2K+1), 3 \times M, 4 \times 4, 4 \times 2K, 4 \times (2K+1), 6 \times 6$ 这几种情况讨论。

86.3 时空复杂度

- 时间：$O(TNM)$；
- 空间：$O(NM)$。

87 OCT11.PARSIN Sine Partition Function

87.1 题目大意

$$f(n,m,X) = \sum_{k_1+k_2+\cdots k_m=N} \sin(k_1X)\sin(k_2X)\cdots\sin(k_mX)$$

给定 m,n,X，计算 $f(n,m,X)$

- $m \le 30$
- $n \le 10^9$

87.2 算法讨论

可以得到递推式 $f(i,j,X) = f(i-1,j-1,X)\sin(X) + 2f(i,j-1,X)\cos(X) - f(i,j-2,X)$，用矩阵乘法计算。

87.3 时空复杂度

- 时间：$O(M^3 \log N)$；
- 空间：$O(M^3)$。

88 AUG11.DIVISORS Something About Divisors

88.1 题目大意

对于给定的正整数 B 和 X，求满足条件的正整数 N 的个数: 要求对于 N，至少存在一个数 $D(N < D \le B)$ 能整除 $N \times X$。T 组数据。

- $T \le 40$
- $X \le 60$
- $B \le 10^{12}$

88.2 算法讨论

设 $i = \frac{NX}{D}$，显然 $i < X$，可以考虑枚举 i，计算对应的满足 $i|NX$ 的 N 的数量。但是可能存在 $i < j < N, j|NX$，导致重复计数。

因为 $i|NX$，有 $\frac{i}{gcd(i,X)}|N$。设 $A_i = \frac{i}{gcd(i,X)}, N = A_i p$，因为 $N \le \frac{B_i}{X}$，p 有上界 $P = \frac{B_i}{XA_i}$。

考虑 $j|A_iXp$，即 $\frac{j}{gcd(j,A_iX)}|p$，设 $B_j = \frac{j}{gcd(j,A_iX)}$，可以枚举 j 的集合容斥计算答案 $ans_i = \sum(-1)^{|S|}\frac{P}{lcm(B_j|j\in S)}$。直接计算会超时，注意到 lcm 超过 P 时贡献为 0，剩下的有效的 lcm 不多，可以用 DP 来计算。

还有一个问题是 $X \in \{58, 59\}$ 时非常慢，有两个优化。一个是如果有 $B_{j_1}|B_{j_2}$ 可以舍去 B_{j_2}。另一个是可以对所有询问排序，X 一样的一起算。

88.3 时空复杂度

- 时间：$O(X^2C)$，C 为有效的 lcm 个数；
- 空间：$O(X^C)$。

89 JULY11.YALOP Trial of Doom

89.1 题目大意

有一个 $N \times M$ 的房间，每个格子是红色或蓝色，起点在 $(1, 1)$，终点在 (N, M)，每离开一个格子，对应的格子和上下左右四格共 5 格颜色取反。移动为八连通。规定到达终点时所有格子必须蓝色，问是否能到达终点。T 组数据。

- $T \le 50$
- $N, M \le 10^9$
- $min(N, M) < 40$
- 红格子数 $K \le 10000$

89.2 算法讨论

不妨设 $M <= N$，当 $M > 1$ 时实际上每次操作可以任意选一格将周围的 5 格反色，因为可以用 2×2 的区域实现“不改变颜色走一格”和“原地改变颜色”两种操作。此时本题相当于普通的“点灯”游戏，可以用 xor 方程组高斯消元求解。本题 N 很大，可以考虑通过递推把所有红色格子转移到第一行。递推有两种实现，一是矩阵乘法，但本题状态大小为 $2M$，矩阵乘法复杂度太高无法承受；二是找循环节，本题循环节长度都不长，可以用这种方法。高斯消元可以压位。

再考虑 $M = 1$ 的情况，可以发现：总步数的奇偶性是一定的；如果前 N 个格子离开次数的奇偶性确定，最后一个格子仅有一种可行的奇偶性；如果第一个格子离开次数奇偶性确定，后面的所有格子都确定。可以枚举第一个格子的两种情况，用 $O(K)$ 的算法计算后面所有格子的状态，判断奇偶性是否有矛盾。实现较为繁琐。

89.3 时空复杂度

- 时间：$M > 1$ 的情况为 $O((K + M)P + M^2)$（P 为循环节长度），$M = 1$ 为 $O(K)$；
- 空间：$O(M^2)$。

90 JUNE11.MINESREV Minesweeper Reversed

90.1 题目大意

倒过来玩扫雷游戏，每次点击关闭一个方块，在正常的扫雷游戏中可能会被一起打开的方块在这里也会被一起关闭。问最少要点几次。

- 数据组数 $T \le 50$
- $R, C \le 50$

90.2 算法讨论

每个雷肯定要点一次，先统计好。考虑没有数字的方块（以下简称空方块）组成的连通块，显然是会一起打开的，打开时也会打开周围的有数字方块。要打开一个空连通块，可以点其中的任意一个空方块，也可以点与他相邻的有数字方块。一个有数字方块最多和两个空连通块相连，这两个空连通块可以一起打开。把空连通块看成点，能一起打开的空连通块之间连边，则最小点击次数为 点数 – 最大匹配，可以用带花树算法。另外，不和空连通块相连的有数字方格是要每个点一下的。

90.3 时空复杂度

- 时间：单组询问最坏 $O(R^3C^3)$，但本题可以通过；
- 空间：$O(RC)$。

Challenge 题：

91 JUNE15.SEADIVM Sereja and Matrix Division

91.1 题目大意

有一个整数矩阵，要求在这个矩阵中选择若干个矩形区域，使被矩形区域包含的数字的可重集合与区域外相等（即每种数字在区域内外出现次数都为总次数的一半），矩形数越少分数越高。100 组数据，每组数据矩阵都为 100×100。

91.2 算法讨论

算法大体分三步：构造初始集合，随机调整，根据集合构造矩形。初始集合可以每种数字随意选“出现次数的一半”个。考虑随机调整，设 $In(x, y)$ 表示 (x, y) 这个位置是否在集合内，如果 $In(x, y) \neq In(x0, y0)$，交换 (x, y) 和 $(x0, y0)$ 的状态，不破坏合法性。判断是否应该交换可以利用估价函数，比较好的一个估价函数是 $h(x, y) = (In(x, y) \oplus In(x, y-1)) + (In(x, y) \oplus (x, y+1))$，$h(x, y) + h(x0, y0)$ 值越大，交换越有利（交换后横向上更聚集），这个过程可以利用“卡时”反复进行。构造完集合后可以直接贪心构造矩形。

这里有一个效果很好的优化：第一步构造初始集合时，首先选一个尽量大的矩形。这样可以避免初始集合过于分散导致结果不优。

91.3 时空复杂度

- 时间：每组数据需限制在 $O(N^3)$ 以内；
- 空间：$O(N^2)$。

92 MAY15.SHOPPING Shopping

92.1 题目大意

有 M 种面值的纸币（最小面值为 1），要处理以下几种事件（强制在线）：

- 付 x 元钱。收银员有无限数量的所有面值的纸币，会贪心找零。
- 到某个 ATM 机取 x 元钱（必须全部取出）。ATM 机有无限数量的纸币，但不同 ATM 机能提供的纸币面值集合不同（都包括 1），ATM 机贪心出钞。
- 银行新发行了一种纸币。
- 城市里新建了一台 ATM 机。

要求使收银员找零的张数尽量少。

92.2 算法讨论

对于取钱，一个直观的想法是面值越小越好，由于总钱数一定，面值越小则张数越多，可以最大化总张数（1 元纸币比较特殊，可以乘一个权值）。实践证明，这种策略效果确实很好。

对于付钱，可以将贪心和背包两种策略并用，有较好的效果。

92.3 时空复杂度

由于策略比较复杂，复杂度难以计算。卡在限制内即可。

93 JAN15.SEAND2 Sereja and Number Division 2

93.1 题目大意

有一个数位中不包含 0 的十进制数 A 和 N 个数 B_i。可以重新排列 A 的数位，要求 $\sum(A \mod B_i)$ 尽量小。

- $A \leq 10^{1000}$
- $N \leq 100$
- $B_i \leq 10^6$
- 数据组数 $T \leq 100$

93.2 算法讨论

由于多个取模函数相加后的对应关系规律性不是很强，本题也比较难以找到比随机更好的方法。B_i 较大的时候只要 random_shuffle 几十次就有较好的效果。B_i 较小时直接这样做效果略差。注意到随机全排列效率很差（因为肯定不能遍历所有排列，而且有很多数位相同，排列这些数位只能浪费时间），可以取出 10 个左右的尽量不同的数位，仅随机这些数位的排列，增大随机次数，使答案更优（需要预处理每一位的贡献）。

93.3 时空复杂度

- 设 $A = 10^L$
- 时间：$O(TN(L + XC))$，X 为选定的数位个数，C 为随机次数；
- 空间：$O(N + L)$。

94 OCT14.CHEFPNT Chef and Painting

94.1 题目大意

有一个 $N \times M$ 的网格，有些格子已经涂上颜色。每次操作可以选择极大的横向或纵向连续一段空格子（即选定一个空格子，向左右（或上下）扩展直到遇到边界或有色格子）涂上颜色。要求步数尽量少。

- $N, M \le 100$
- 初始有色格子 $K \le 3000$

94.2 算法讨论

可以考虑用贪心法，每次涂上最长的一段，有多个最长的就随机选一个（也可以额外引入一些随机性），多随机几次。为了减少单次随机贪心的复杂度，需要对每一行以及每一列开一个线段树来维护。

94.3 时空复杂度

- 时间：$O(RndCnt \times NM \log(N+M))$；
- 空间：$O(NM)$。

95 SEPT14.FACTORIZ Factorisation

95.1 题目大意

给定一个数 N，要求输出 M 个数，它们乘积为 N。

- 100 组数据，得分为 M 的平方和。
- 10 组,$N \le 10^{18}$，随机均匀分布
- 15 组,$N \le 10^{18}$，不是随机均匀分布
- 50 组,$N \le 10^{1000}$，随机均匀分布
- 25 组,$N \le 10^{1000}$，所有质因子不超过 10^{18}

95.2 算法讨论

前 25 组数据表面上可以用 Pollard-rho 算法分解质因数。但考虑大于 100 的质因子，这些质因子在一个数中小于 9 个，分解出这些质因子不但耗费大量时间而且对答案的贡献不多。所以对于这些数据不要分解质因数，直接取几个小质数试除即可。

考虑 N 比较大的情况，这里面有的数容易分解，有的数难以分解。一种效果较好的做法是，先取小质数（前 100 ～ 200 个）试除，找到已分出质因数最多的若干个数（15 个左右），用较大的质数（1200000 以内）试除。注意取模的常数。后一个试除中可以先枚举质数，预处理每一位取模的贡献，然后对所有的数试除，大幅减少取模次数。

95.3 时空复杂度

- 设 $N = 10^L$，C 为小质数试除的质数个数，M 和 Q 分别为较大的质数的范围和个数，K 为选定用较大质数试除的数的个数。
- 时间：$O((TC + KQ)L \log L)$；
- 空间：$O(L + M)$。

96 AUG13.DELNMS Deleting numbers

96.1 题目大意

有 N 个数，每次可以选择两个数 v,t，要保证 A_{v+kt} 都相等，并把这些数删除（下标会变化）。要求用尽量少的操作删除所有数。$N \leq 100000$

96.2 算法讨论

仅出现一次的数每个肯定要一次操作，可以先全删除掉。一种简单的方法是，保留出现最多的数，然后其它每个数从后到前用一次操作删除，然后再用一次操作删除出现最多的数。这样效果不是很好。可以考虑删除其它数时每次多删几个，比如，若存在最近的 $A_i = A_{i-t}$，将它们一起删除，再检查一下 $A_{i-2t}, A_{i-3t}, \cdots$ 这些是否等于 A_i，一并删除末端的连续一段。整个过程中需要用树状数组维护下标。

96.3 时空复杂度

- 时间：$O(N \log N)$；
- 空间：$O(N)$。

97 JUNE13.CHAORNOT To challenge or not

97.1 题目大意

有 N 个不同的数，选出尽量多的不包含等差数列的数。$N, A_i \leq 100000$

97.2 算法讨论

最终选出来的数不会太多。考虑维护一个 set，里面包含禁止选择的数。依次扫描每个数判断是否能选，每次选出一个数后更新这个 set。如果一个数能选，可以以一定概率不选（不选概率在 $\frac{1}{70}$ 左右效果较好），利用随机性构造多种解，取最优的那一个。

97.3 时空复杂度

- 时间：$O(RndCnt \times NM \log M)$，M 为最多选出的数的数量；
- 空间：$O(N)$。

98 OCT12.MAXRECT Maximum Sub-rectangle in Matrix

98.1 题目大意

有一个 $N \times M$ 的整数矩阵，找出一个数字和尽量大（必须为正）的“子矩阵”。这里的“子矩阵”定义与平时不同，定义为：选出若干行和列，取行列相交处的所有元素。

- $N, M \leq 300$

98.2 算法讨论

考虑一种类似“迭代法”的方法：先随机选若干列，再根据列选最优的行，再根据行选最优的列，……这样做可以得到很优的解。

98.3 时空复杂度

- 时间：$O(NMC)$，C 为迭代次数；
- 空间：$O(NM)$。

99 JUNE12.CLOSEST Closest Points

99.1 题目大意

三维空间里有 N 个点。Q 个询问，每次询问离某个坐标的最近点。不要求全部答对，按答对的个数给分。

99.2 算法讨论

最近邻问题是 K-d 树的经典应用。实现良好的 K-d 树能够正确回答清橙测试数据中的全部询问。

99.3 时空复杂度

- 时间：$O((N+Q)\log N)$；
- 空间：$O(N)$。

100 MAY12.CKROACH Killing Gs

100.1 题目大意

有 N 只虫子，M 种杀虫剂。第 j 种杀虫剂价格为 C_j，对所有虫子使用之后第 i 只虫子死亡的概率为 P_{ij}（一种杀虫剂只能用一次）。现在要使所有虫子死亡的概率至少为 90%，要求给出一个总价格尽量少的方案。$N, M \leq 200$

100.2 算法讨论

考虑 $N = 1$ 的情况，设 $W_i = \log(1 - P_{i,1})$，则目标是 $\sum A_i W_i \leq \log \frac{1}{10}$，其中 $A_i \in \{0, 1\}$，最小化 $\sum A_i C_i$。这是一个 01 背包问题。说明这道题即使是 $N = 1$ 的情况也是很难得到最优解的。可以考虑使用贪心算法或随机化算法。直接按照性价比贪心可以有较好的效果，但难以进一步优化。随机化算法中可以使用模拟退火算法，在选择了合适的邻域和参数后效果非常好。

100.3 时空复杂度

- 时间：$O(RndCnt \times NM)$；
- 空间：$O(NM)$。