IOI2015 中国国家集训队第一次作业第二部分试题泛做

Chengchi Zhou

2016 年 1 月 19 日

目录

JUNE11.CLONES	**6**
JUNE11.MINESREV	**6**
JULY11.BB	**6**
JULY11.YALOP	**8**
JULY11.KITCHEN	**9**
AUG11.SHORTCIR	**10**
AUG11.DIVISORS	**11**
SEP11.SHORT	**11**
SEP11.CNTHEX	**11**
OCT11.BAKE	**11**
OCT11.PARSIN	**12**
NOV11.LUCKYDAY	**12**
NOV11.DOMNOCUT	**13**
DEC11.SHORT2	**13**
DEC11.HYPER	**13**
JAN12.CARDSHUF	**14**
JAN12.MISINT2	**14**
FEB12.FLYDIST	**14**
FEB12.FINDSEQ	**15**
MARCH12.EVILBOOK	**15**
MARCH12.CIELQUAK	**15**
APRIL12.TSUBSTR	**16**
APRIL12.CONNECT	**16**
APRIL12.SIMGRAPH	**16**
MAY12.TICKETS	**16**
MAY12.LEBOXES	**17**

JUNE12.MATCH 17

JUNE12.COOLNUM 17

JUNE12.CLOSEST 18

JULY12.DGCD 18

JULY12.EST 18

AUG12.MAGIC 18

AUG12.GTHRONES 19

SEPT12.KNGHTMOV 19

SEPT12.SIMNIM 19

OCT12.MAXCIR 20

OCT12.MAXRECT 20

NOV12.COUNTARI 20

NOV12.MARTARTS 20

DEC12.DIFTRIP 21

JAN13.ANDOOR 21

JAN13.CUCUMBER 21

FEB13.ROC 22

FEB13.QUERY 22

MARCH13.LECOINS 22

MAY13.QTREE 22

JUNE13.TKCONVEX 23

JUNE13.SPMATRIX 23

JUNE13.CHAORNOT 23

JULY13.RIVPILE 24

AUG13.LYRC 24

AUG13.PRIMEDST 24

AUG13.DELNMS 24

SEP13.TWOROADS 24

OCT13.FN 25

NOV13.MONOPLOY 25

NOV13.QPOINT 25

DEC13.QTREE6 26

DEC13.REALSET 26

JAN14.CNTDSETS 26

JAN14.TAPAIR 27

FEB14.DAGCH 27

FEB14.COT5 27

MARCH14.GERALD07 28

MARCH14.STREETTA 28

APRIL14.GERALD08 28

MAY14.ANUDTQ 29

MAY14.SEINC 29

JUNE14.TWOCOMP 29

JUNE14.SEAARC 30

JULY14.GNUM 30

JULY14.SEAEQ 30

AUG14.SIGFIB 30

AUG14.PUSHFLOW 31

SEPT14.QRECT 31

SEPT14.FIBTREE 31

OCT14.TRIPS 32

OCT14.BTREE 32

OCT14.CHEFPNT 32

NOV14.FNCS 33

NOV14.SEAORD 33

DEC14.DIVIDEN 33

DEC14.DIVIDEN 34

DEC14.RIN 34

JAN15.RANKA 34

JAN15.XRQRS 34

JAN15.SEARD2 35

FEB15.DEVLOCK 35

FEB15.CUSTPRIM 35

MARCH15.TREECNT2 36

MARCH15.RNG 36

APRIL15.BWGAME 36

APRIL15.LPARTY 36

MAY15.CBAL 37

MAY15.GRAPHCNT 37

JUNE15.CHEFBOOK 37

JULY15.EASYEX 38

JULY15.HAMILG 38

AUG15.CLOWAY 38

AUG15.DISTNUM 39

JUNE11.CLONES

题目大意

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。

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 $f(0, \cdots, 0) = 0$

P 是 1-保留值函数集合: 满足 $f(1, \cdots, 1) = 1$

D 是自对偶函数集合: 满足 $!f(x_1, \cdots, x_n) = f(!x_1, \cdots, !x_n)$

A 是仿射函数集合: 满足如果 $f(a_1, \cdots, c, \cdots, a_n) = f(a_1, \cdots, d, \cdots, a_n)$ 则 $f(b_1, \cdots, c, \cdots, b_n) = f(b_1, \cdots, d, \cdots, b_n)$ 的函数

求这几个集合的组合中有多少个 N 项函数。

数据规模：$1 \le N, Q \le 100$。

算法讨论

四中 clone 能组出 16 种不同的组合，预处理出这 16 种情况下的答案。在表达式计算时用一个二进制数来记录这 16 种情况是否在当前集合中。最终可以得到整个表达式的二进制数，直接累加即可。

时间复杂度 $O(T \mid S \mid)$。

JUNE11.MINESREV

题目大意

给出一个 $R \times C$ 的扫雷网格，雷的位置已知，现在要通过点击一些格子来打开所有的格子，求最少的点击次数。

数据规模：$1 \le T \le 50, 1 \le R, C \le 50$。

算法讨论

每个雷显然都需要一次点击来打开。剩余的格子分成两类，第一类格子相邻格中没有雷，第二类有雷。第二类格子会组成一些连通块。点击连通块中的一个格子就可以打开整个连通块和连通块旁边的第一类格子。发现优先第一类格子无法被第二类格子打开，就需要单独点击。现在考虑打开所有的连通块。这可以通过点击连通块内的点和连通块周围的第二类格子，而第二类格子有可能可以打开两个连通块。于是按打开情况建图，问题装化为一般图的最大匹配，使用带花树解决。

时间复杂度 $O(TR^3C^3)$。

JULY11.BB

题目大意

计算长度为 N 的 01 串个数，需要满足任意连续 M 个位置中至少有 K 个 1，并且 1 的个数最少。

数据规模：$1 \le K \le M \le 50$，$M \le N \le 10^9$。

算法讨论

首先考虑 M 整除 N 的情况，可以把串分成 N/M 段，每段长度为 M。那么每一段中至少有 K 个 1，整个串至少有 $K * (N/M)$ 个 1。在每一段的最后 K 个位置放 1 所得到的串是符合条件的，那么可以证明 1 最少的个数是 $K * (N/M)$。

观察下面这个例子:

$$N = 16 \quad M = 4 \quad K = 2$$

0011 0101 1010 1100

JULY11.BB continued on next page...

可以发现对应 1 在段中的位置在逐渐减小。段中第一个 1 的位置从 3 到 2 到 1，第二个 1 的位置从 4 到 4 到 2。把这写成矩阵：

3 2 1 1

4 4 3 2

这就是一个半标准杨氏矩阵。方案数计算公式如下：

$$dim = \prod_{(i,j)} \frac{r+j-i}{hook(i,j)}$$

其中 $hook(i,j)$ 表示矩阵中同一行或同一列，纵坐标或横坐标大于等于 (i,j) 的个数。r 表示候选数集合大小。

同样对于上面这个例子，分子部分：

4 5 6 7

3 4 5 6

分母部分：

5 4 3 2

4 3 2 1

可以发现上下形式相同，当 N 很大的时候存在大量可以抵消的数字。有效数个数是 $O(K*M)$。

考虑 M 不整除 N 的情况，如果 $N\%M <= M-K$，那么每一组的前 $N\%M$ 个位置都应该是 0。如果 $N\%M >= M-K$，那么每一组最后 $M-N\%M$ 的位置都应该是 1。这个可以用相同方式解决。

时间复杂度 $O(K*M*logMod)$。

JULY11.YALOP

题目大意

约翰尼进入了一个大房间，这个房间被划分成了 $n*m$ 个方格，有 k 个方格是红色，其余都为蓝色。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方格都是蓝色的。约翰尼可以移动到八个相邻的方格上，每当他离开一个方格，那么这个方格和它周围的四个方格会改变颜色。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。

数据规模：$1 \le t \le 50$，$1 \le n,m \le 10^9$，$min(n,m) < 40$，$0 \le k \le min(m*n, 10000)$，$1 \le x \le n$，$1 \le y \le m$

算法讨论

一个方格经过奇数次才有效，那么可以用 $[0,1]$ 来表示这个方格是否有效，所有方格组成的二进制数可以看成这种移动路径的状态。如果任意一种移动状态都是可以得到的，那么具体的移动方式就不需要再考虑，只需要知道是否存在一种二进制状态使得所有方格都变成蓝色。

下面证明任意一种移动状态都是可行的。

考虑一个 $2*2$ 的方格组，由于移动的连通方式是八连通，这 4 个方格是两两可达的，可以用长为 4 的二进制数来表示方格状态。

操作一：在不改变方格状态的情况下，从当前方格移动到任意一个方格。

以从第一个方格移动到第二个方格为例，粗体表示当前所在方格。

0000

10**0**0

1010

00**1**0

0**0**00

操作二：改变当前所在方格的颜色。

这个在操作一的基础上很方便，以第一个方格为例。

0000

1**0**00

1000

有上述两种操作，对于 $2*2$ 的方格组，可以将这个方格组转化成任意一种状态并移动到任意一个方格上。考虑一个 $n*m(min(n,m) > 1)$ 的方格组，可以把它划分成一些 $2*2$ 的方格组，运用上述两种操作就可以将 $n*m$ 的方格组转化成任意一种状态并移动到任意一个格子上。最后会讨论 $min(n,m) = 1$ 的特殊情况。

考虑 $min(n,m) > 1$,$n \le m$ 的情况。对于 (x,y) 这个方格，如果它是红色的，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，可以通过循环节来计算 y 比较大的情况。红色方格的异或值就是需要达到的状态，变量就是最后一列的状态。这可以通过线性基或是高斯消元来解决。

考虑 $n = 1$,$n \le m$ 的情况，先不考虑往回走，那么只能从 $(1,1)$ 走到 $(1,m)$。往回走和返回的步数和一定是偶数，计算出步数判断是否为偶数就可以了。

时间复杂度：$O(k + \min(n,m) * Len)$，Len 表示循环节大小。

JULY11.KITCHEN

题目大意

给出一个 $N * M$ 的全白方格图，你需要在某些方格中填入黑色，并满足如下条件：

1. 黑色方格是四连通的。

2. 不存在回路，即不存在这样的一组黑色方格 $X_1, X_2, X_3, \ldots, X_n$，$n > 3$，满足 $X_1 = X_n$，且对于任意 k,$1 \leq k < n$，X_k 和 X_{k+1} 相邻。

数据规模：$T <= 30$ $N, M <= 100$

算法讨论

问题就是在方格图种放入一棵树，使得树的节点最小。

首先可以得到一个比较暴力的做法，随机一个点，以这个点为根向外延生从而生成这棵树。可以多次随机取最优值，但这样的做法无法得到很好的据结果。

考虑构造，用 V 表示树的节点数，E 表示树的边数。如果把方格全染黑，那么

$$V = N * M$$

$$E = N * (M - 1) + M * (N - 1)$$

而这显然是不合法的。我们需要让其中 K 个格子回到白色。用简单的容斥可以得到

$$V = N * M - K$$

$$E = N * (M - 1) + M * (N - 1) - 4K + Border + 2Corner + Neigh$$

其中 $Border$ 表示在边界上的白格个数，$Corner$ 表示在角落的白格个数，$Neigh$ 表示相邻白格对数。而对于树而言，$V = E + 1$，可以得到

$$K = \frac{(N - 1) * (M - 1) + Border + 2Corner + Neigh}{3}$$

为了让白格最多，也就是 K 最少，就需要让 $Border + 2Corner + Neigh$ 最小。那么基本的想法就是构造出一组较为优的解，并根据 N、M 的情况在局部进行微调。

这里给出一个基本解：

##########

#.#.#.#.#.

##.#.#.#.#

.#########

##.#.#.#.#

#.#.#.#.#.

##########

#.#.#.#.#.

##.#.#.#.#

.#########

可以发现遵循这种填充方式在一些情况会出现不连通，需要进行调整使其连通。

AUG11.SHORTCIR

题目大意

给定包含 n 个未知数、and、or、not 和括号的表达式。第 i 个未知数有 p_i 的概率是 1，否则为 0。现在想要确定这个表达式的值，每次操作可以查询一个变量的值，求确定表达式值的期望查询次数。

数据规模：$1 \le Len \le 30000$，$1 \le n \le 1000$。

算法讨论

先考虑建出这个表达式的语法树。如果不考虑 not 的影响，可以用 $Tree_{l,r}$ 来表示将 $l \sim r$ 的表达式建成树后的情况，再根据表达式的情况进行递归。

有恒等式：

$$\text{not(x and y) = (not x) or (not y)}$$

$$\text{not(x or y) = (not x) and (not y)}$$

那么可以把 not 当做一个标记，沿着树下传并改变运算符。

建出语法树后可以用 dp 计算答案。用 P_i 表示第 i 个节点所代表的子树权值为 1 的概率，T_i 表示得到第 i 个节点所代表的子树权值的期望步数。

以 $X = \text{and}(son_1, son_2, \ldots, son_k)$ 为例：

$$P_X = P_{son_1} \times P_{son_2} \times \cdots \times P_{son_k}$$

$$T_X = T_{son_1} + P_{son_1} \times (T_{son_2} + P_{son_2} \times (T_{son_3} + \cdots))$$

son_i 在 son_j 之前需要满足：

$$T_{son_i} + P_{son_i} \times T_{son_j} < T_{son_j} + P_{son_j} \times T_{son_i}$$

即

$$T_{son_i}/(1 - P_{son_i}) < T_{son_j}/(1 - P_{son_j})$$

将儿子节点按照这个关系排序后就可以计算了。or 同理。

AUG11.DIVISORS

题目大意

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

数据规模：$1 \leq T \leq 40, 1 \leq X \leq 60, B \leq 10^{12}$。

算法讨论

令正整数 $i = \frac{NX}{D}$，那么有 $i < X$，枚举每一个 i，计算满足 $i \mid NX$ 且不存在 j 满足 $i < j < X$ 且 $j \mid NX$ 的 N。由于 $i \mid NX$，所以有 $\frac{i}{gcd(i,X)} \mid N$，令 $A_i = \frac{i}{gcd(i,X)}$，那么有 $N = A_i p$,，由于 $N \leq \frac{B_i}{X}$，所以有 $p \leq \frac{B_i}{XA_i}$。

由于 $j \mid A_i X p$, 所以有 $\frac{j}{gcd(A_i X, j)} \mid M$。令 $B_j = \frac{j}{gcd(AiX, j)}$, 枚举 i 再进行容斥, $res_i = \sum (-1)^t \frac{P}{lcm B_j}$，需要注意一些常数优化。

时间复杂度 $O(X^2 K)$。

SEP11.SHORT

题目大意

给定 n,l，求满足 $n < a < l, n < b < l, (a-n)(b-n) \mid ab - n$ 的 (a, b) 组数。

数据规模：$0 \leq n \leq 10^5$，$l \leq 10^{18}$。

算法讨论

令 $p = a - n$,$q = b - n$，问题转化为 $1 \leq p < l - n, 1 \leq q < l - n, pq \mid (p+n)(q+n) - n$。设 $k = \frac{(p+n)(q+n)-n}{pq}$, 则可以得到

$$q = \frac{n(p+n-1)}{kp - p - n}$$

当 p 过大时无解，枚举 p。当 p 较小时可以暴力枚举 $n(p+n-1)$ 的质因数来得到 q，当 p 较大时枚举 k 来得到 q。若 q 在要求范围内就得到了两 (一) 组解。

SEP11.CNTHEX

题目大意

有长度为 1 至 N 的木棍，每种木棍都有 K 根，需要从中选出六根木棍使得能拼出面积为正的六边形。选出的木棍需要满足最长的木棍长度不超过 L，剩余木棍不超过 X。求方案数。

数据规模：$1 \leq L \leq 5, 1 \leq X < L \leq N \leq 10^9, N - L \leq 100$。

算法讨论

首先枚举最长一根木棍的长度，其余的木棍需要满足长度和大于最长的木棍长度。

由于长度范围很大，考虑用数位 dp 计算。用 dp[i][j][k][l][p] 表示当前处在后 i 位，其余木棍长度和和最长木棍的关系是 j，其余木棍的大小关系是 k，和 X 的关系是 l，对后 i+1 位的进位是 p 的方案数，其中关系可以用二进制数来表示。转移需要枚举各个木棍下一位的数值。

时间复杂度 $O((N-L) \log NState)$，注意去除无效状态。

OCT11.BAKE

题目大意

支持两种操作，第一种操作加入一个新顾客，第二种操作询问满足某种条件的顾客消费数。

算法讨论

OCT11.BAKE continued on next page...

用数组维护，询问时枚举年龄即可。时间复杂度 $O(m*age)$。空间复杂度 $O(Product*Size*Province*City*Region*age)$。

OCT11.PARSIN

题目大意

给定 M,N,K，计算

$$\sum_{k_1+k_2+\cdots+k_M=N} \sin(k_1X)\sin(k_2X)\cdots\sin(k_MX)$$

$$k_1,\ldots,k_M\text{是非负整数}$$

数据规模：$1 \le M \le 30$，$1 \le N \le 10^9$，$0 \le X \le 6.28 < 2\pi$。

算法讨论

$$\sin kX = 2\cos X\sin(k-1)X - \sin(k-2)X$$

由此可以得到：

$$\sin k_1X\sin k_2X\cdots\sin k_mX =$$

$$2\cos X(\sin k_1X\sin k_2X\cdots\sin(k_m-1)X - \sin k_1X\sin k_2X\cdots\sin(k_m-2)X$$

$$f(m,n,X) = \sin Xf(m-1,n-1,X) + 2\cos Xf(m,n-1,X) - f(m,n-2,X)$$

这是一个线性递推，使用矩阵乘法即可。

时间复杂度 $O(M^3\log N)$。

NOV11.LUCKYDAY

题目大意

定义

$$S_1 = A$$

$$S_2 = B$$

$$S_i = (XS_{i-1} + YS_{i-2} + Z)\ mod\ P\ for\ i \ge 3$$

有 Q 个询问，对于每个询问 (L,R)，求满足 $L \le k \le R$，$S_k = C$ 的个数。

数据规模：$2 \le P \le 10007$, P 是质数, $0 \le A,B,X,Y,Z,C < P$, $1 \le Q \le 20000$, $1 \le L \le R \le 10^{18}$

算法讨论

把相邻两个数看成一种状态，那么最多有 P^2 种状态，和 C 有关的状态最多有 P 种。需要算出每一种与 C 有关状态的起始位置和最短循环节长度。令转移矩阵为 Tr。循环节长度 T 即满足 $ITr^T = I$，用 BSGS 来找最小循环节长度。找起始位置 S 即满足 $ITr^S = G$，G 为目标状态，同样可以用 BSGS 解决。答案就是：

$$\sum \lfloor \frac{L-S_i}{T} \rfloor + 1$$

$$= \sum \frac{(L-S_i) - (L-S_i)\ mod\ T}{T}$$

把 S 排序后二分就可以找到答案了。

需要注意的是 $det(Tr) = -Y$，当 $Y = 0$ 的时候不存在逆矩阵需要特判。在这种情况中是一位递推，状态数最多只有 P 种，可以暴力找。

NOV11.DOMNOCUT

题目大意

给出 N 行 M 列的矩形棋盘。棋盘染色是指：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，且这条直线不能穿过一对有相同颜色的相邻格子。

需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，使得使用的颜色最少。

数据规模：$1 \leq T \leq 3000$，P 是质数，$1 \leq N, M \leq 500$。

算法讨论

分以下几种情况讨论：

1.$N * M$ 是奇数。

2.$N = 1, M = 2k$。

3.$N = 2, M = 2k$。

4.$N = 2, M = 2k + 1$。

5.$N = 3, M = 4$。

6.$N = 3, M = 2k(k > 2)$。

7.$N = 4, M = 4$。

8.$N = 4, M = 2k(k > 2)$。

9.$N = 4, M = 2k + 1$。

10.$N = 5, M = 6$。

11.$N = 5 + 2u, M = 6 + 2v$，这种情况能够以每次加入两行 (列) 的方式扩展情况 10 得到。

12.$N = 6, M = 8$。

13.$N = 6 + 2u, M = 8 + 2v$，这种情况能够以每次加入两行 (列) 的方式扩展情况 12 得到。

14.$N = 6, M = 6$。

可以使用搜索或 dp 来寻找 N，M 较小时的解。

时间复杂度 $O(N * M)$。

DEC11.SHORT2

题目大意

询问有多少对 (a, b)，满足 $(a - p)(b - p) \mid ab$。

数据规模：$1 \leq T \leq 5, 1 \leq p \leq 10^{12}$，p 为质数。

算法讨论

原问题等价于 $ab \mid p(a + b + p)$ 的对数。

分情况讨论：

1.$a \mid p, b \mid p$，只有 5 组解。

2.$a \nmid p, b \nmid p$，令 $a < b$，此时有 $b = \frac{a+p}{ak-1}$，需要满足 $p \nmid a$，$(ak - 1) \mid (a + p)$，$a \mid ak$。枚举后判断。

3.$a \mid p$ 或 $b \mid p$，对数是第二种情况的两倍。

时间复杂度：$O(T\sqrt{p})$。

DEC11.HYPER

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都以后都不连通的 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.

数据规模：$1 \leq T \leq 15, 3 \leq N$。

算法讨论

可以将一棵超树分成若干个点双联通分量，对于每一个分量都有去掉任意一个点超树依然联通。对于一棵点双联通的超树，每一条边连接的三个点中都恰好有一个点为叶子节点。删去该点后，就得到了一个常规的点双连通图，原来的超图的点数对应着这个图的点数加边数。

使用搜索求出点数为 d 的双联通超树数量。接着通过搜索合并这些点双联通超树。

时间复杂度：$O(T)$。

JAN12.CARDSHUF

题目大意

初始一个长为 N 的序列，有 M 个操作，需要支持移动区间，把区间逆序。

数据规模：$1 \leq N, M \leq 100000$。

算法讨论

使用平衡树维护这个序列。以 splay 为例，移动区间可以先取出这个区间，再插入到合适位置。区间逆序可以加逆序标记，在适当时候下推标记。

时间复杂度 $O(M \log N)$。

JAN12.MISINT2

题目大意

一个长度为 N 的字符串 S，如果把它的偶数位依次写到开头，再依次写奇数位，得到的新串和原串相同，那么称它是好的。求长度在 L 到 R 的字符串中好串个数。

数据规模：$1 \leq L, R \leq 10^{10}$ $R - L \leq 5 \times 10^4$。

算法讨论

令长度为 n 的字符串的置换中的循环数量是 $f(n)$，那么答案就是 $\sum_{i=L}^{R} f(i)$。令 $ord(mod)$ 为 2 模 mod 的阶，可以得到 $f(n) = \sum_{p|(n+1),p>1} \frac{\phi(p)}{ord(p)}$。求 $\phi(p)$ 可以对 $[L, R]$ 中的数分解质因数后直接计算。需要求 $ord(x)$，发现当 a、b 互质时有 $ord(ab) = lcm(ord(a), ord(b))$，那么只需要计算出所有 $ord(p^k)$ 的答案，再每次合并就可以了。

时间复杂度 $O(T(R-L)(\frac{\sqrt{R}}{\log R} + \log^2 R))$。

FEB12.FLYDIST

题目大意

有 N 个城市，在这些城市间有 M 条直达航线。厨师将第 i 条航线的距离增加或减少改变 d_i 需要付出 d_i 的金额。使得在改变之后，没有一条航线的距离比经过中转点的距离还要长。厨师需要求出最少花费的金额数。

数据规模：$1 \leq N \leq 10, 1 \leq M \leq \frac{N*(N-1)}{2}$。

算法讨论

对于初始边长 $Slen_i$,若经过操作后它的长度变为 $Tlen_i$,那么所需要的花费就是 $\sum |Slen_i - TLen_i|$。绝对值的运算相对麻烦，可以设 Up_i、$Down_i$，满足 $SLen_i + Up_i - Down_i = TLen_i$。花费就是 $\sum Up_i + Down_i$。

考虑其他约束条件，合法方案中每条边的长度都是这两点在图中的最短路，即 $dist_{i,j} \leq dist_{i,k} + dist_{k,j}$。

问题转化为一个线性规划。但由于限制中的常数项不一定为 0，导致变量都为 0 是不合法的初始状态。观察方程发现常数项都是关于 Len_i 的, 且正负号与 $Down_i$ 相反, 可以令 $NDown_i = Len_i - Down_i$，这样消除了常数项，$NDown_i$ 的限制是 $NDown_i \le Len_i$。

时间复杂度 O(单纯形复杂度)。

FEB12.FINDSEQ

题目大意

给出 N 个数 $A_1, A_2, \ldots, A_n$，和一个 5 个数的排列 B。要在 N 个数中找出 5 个数使得这 5 个数的大小关系和排列 B 相同。

数据规模：$1 \le N \le 1000$。

算法讨论

枚举 B_2, B_4 所在位置，数列 A 被分割成 5 个部分 P_1, P_2, P_3, P_4, P_5。

在 P_2, P_4 上的数已经知道了。

在 P_1, P_5 内的数需要知道它两边的某个数 (初始状态下 0,6 为已知数)，以 $B_1 = 2, B_2 = 1$ 为例，首先可以得到 1 所对应的数 c_1，那么 2 由于 1 已知，只需要在 P_1 内找比 c_1 大的最小数，这是可以预处理的。

在 P_3 内的数需要知道它两边的数。由于 P_3 不是一个前 (后) 缀，所以不能直接通过预处理得到大于某个数的最小数，只能得到在某个区间中的数是否存在。得到两边的数后就确定了这个区间，如果不存在就不需要找这个数了。

时间复杂度 $O(N^2)$。

MARCH12.EVILBOOK

题目大意

有 n 个人，打败第 i 个人的代价是 c_i，并得到 d_i 的魔法值。对第 i 个人使用魔法可以使 $c_i = \frac{c_i}{3}$，$d_i = \frac{d_i}{3}$，需要消耗 X 的魔法，求使魔法值大于 666 所需要的最小代价。

数据规模：$1 \le n \le 10$，$10 \le X \le 666$，$0 \le c_i \le 10^9$，$0 \le d_i \le 10^9$。

算法讨论

由于当前魔法值大于 666 就结束了，所以在有魔法的情况下一定会把一个人的魔法值减少到 666 以下。同时打败一个人要有意义，也就是魔法值要增加，所以 $kX < \frac{d_i}{3^k}$，这样每个人的可行次数很少。

既然打败一个人一定可以使得魔法值上升，一定会优先打败需要次数少的人。可以按这个次序搜索，并加入适当地剪枝。

时间复杂度 $O(T4^n)$。

MARCH12.CIELQUAK

题目大意

给出一个 $R * C$ 的完整网格，每条边有 p 的概率损坏，询问点 $(1,1)$ 和点 (n,m) 连通的概率。

数据规模：$1 \le T \le 50, 1 \le R \le 8, 1 \le C \le 10^{18}$。

算法讨论

R 比较小，考虑使用轮廓线 dp，状态可以用最小表示法表示，可知所有可达状态不到 4000 种。当由于 C 非常大，不能直接 dp 得到。

用 $Ans_{r,c,p}$ 来表示答案。当 c 较大时，可以选择一个 x，用 $(\frac{Ans_{r,x,p}}{Ans_{r,x-1,p}})^{c-x}$ 来代替 $Ans_{r,c,p}$。

时间复杂度 $O(T \times R \times x \times State)$。

APRIL12.TSUBSTR

题目大意

给出一个字母树，子串的定义是从某个点到它子树中某个点的字符串，求不同子串个数。每次询问第 k 小子串。

数据规模：$1 \le N \le 250000$，$1 \le Q \le 50000$，$1 \le k < 2^{63}$。

算法讨论

建出后缀自动机，递推算出从某个节点出发可接受的子串个数。初始状态的个数就是不同子串数。寻找第 k 小子串可以从初始状态出发沿着后缀自动机走一遍。

时间复杂度 $O(m \log n)$。

APRIL12.CONNECT

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 $A_{i,j}$，权值为 $V_{i,j}$。现在需要在矩阵中找到一个连通块，这个连通块至少包含 K 中不同的颜色，求最小权值和。

数据规模：$1 \le N, M \le 15, 1 \le K \le 7$。

算法讨论

当矩阵中只有 K 中不同的颜色时问题就是斯坦纳树，用 $dp[i][j][k]$ 表示位于矩阵的 i 行 j 列，连通块中颜色状态为 k 的最小权值和。当有更多的颜色时，每次给每种颜色随机一种 $[1, K]$ 范围内的颜色，使用斯坦纳树计算。多次随机可以保证很高的成功率。

时间复杂度 $O(T2^K NM)$，T 表示随机次数。

APRIL12.SIMGRAPH

题目大意

给出两个 N 个点的图，求两个排列，图中的点编号置换后相同的边最多。

数据规模：$30 \le N \le 75$。

算法讨论

第一个图的排列直接设成 $1, 2, \cdots, N$，随机第二个排列，可以使用暴力随机交换两个数的方式模拟退火。

时间复杂度 $O(TN)$，T 表示模拟退火次数。

MAY12.TICKETS

题目大意

m 个人有可能来参加晚餐，大厨准备了 n 道菜，他要给来的人每人一道菜，第 i 个人吃到第 a_i 或 b_i 道菜就会满足。求最大的 k 使得任意 k 个人来都可以满足。

数据规模：$2 \le n \le 200$，$0 \le m \le 500$。

算法讨论

把菜当做点，人当做边连接 a_i 和 b_i。问题就是求一个最大的子图，它的边数是点数加一。这个可以看成树加上两条边后再去掉和环无关的多余部分。

分两种情况考虑：

1. 两个环相交，这个可以枚举交点，判断交点之间是否存在三条路径，并且长度之和最小。用 bfs 计算。

MAY12.TICKETS continued on next page...

2. 两个环相互独立，它们之间用一个点或一条链连接。先枚举中间点，把它当做根。由于要长度最小，可以建 bfs 树。一个环的大小可以定义为环中点得个数加上环的顶点到根的距离。找最小两个环作为答案。

时间复杂度 $O(n^2m)$。

MAY12.LEBOXES

题目大意

有 n 个盒子，打开第 i 个盒子有 P_i 的概率是 V_i 块钱，$1-P_i$ 的概率是一块钻石。有 m 件物品，购买第 i 件需要花费 C_i 块钱和 D_i 个钻石。打开所有盒子之后去购买最多的物品，求物品件数的期望值。数据规模：$1 \le n, m, D_i \le 30$，$1 \le V_i, C_i \le 10^7$。

算法讨论

先用背包预处理出得到 i 块钻石买 j 件物品所需的最少钱数。考虑使用 meet in middle，先枚举前 x 个盒子的情况，在枚举后 $n-x$ 个盒子的情况。如果枚举后面盒子的情况得到了 a 块钱、b 块钻石，枚举总共有 B 块钻石，买了 c 件物品，那么根据预处理可以知道在前面的盒子中需要的钻石数和钱数，在前面的统计结果中二分就可以得到答案了。

时间复杂度 $O(x2^x + xn^2 2^{n-x})$。

JUNE12.MATCH

题目大意

给出一个两边点数分别为 N,M 的二分图，左边的第 i 个点和右边的第 j 个点有边相连的概率为 $p_{i,j}$。求二分图最大匹配的期望值。

数据规模：$1 \le N \le 5$，$1 \le M \le 100$。

算法讨论

定义状态 $state$ 为长度为 5 的二进制数，表示所在位为 1 的左边点集合可以找到匹配。将 $state$ 状压得到 $STATE$，表示所在位为 1 的 $state$ 可行。如果存在一个可行 $state$，那么这个 $state$ 的子集显然也是可行的。由此搜索可以得到所有可行的 $STATE$ 只有 7000 多种。于是可以用 $dp_{i,j}$ 来表示右边的第 i 个点完成匹配，当前的 $State$ 为 j 的概率。转移即枚举这个点与左边的哪个点匹配。最后统计答案。

时间复杂度 $O(2^N * M * STATE\_NUMBER)$。

JUNE12.COOLNUM

题目大意

一个数有 k 位，每个数位上的数字分别为 $X_1, X_2, \ldots, X_K$，如果一个数 n 存在一到三个数位上的数的和为 s，且 $(X_1 + X_2 + ... + X_K - S)S$ 是 n 的倍数，那么 n 是 cool number。定义 L_N 和 R_N 分别是小于等于 N 的最大的 cool number 和大于 N 的最小的 cool number，多次询问给定 N，求 L_N 和 R_N。

数据规模：$1 \le T \le 10^5$，$1 \le N \le 10^{1000}$。

算法讨论

如果数 n 只有小于三个数位有数, 那么定义这个数为第一类 cool number, 可知其余的 cool number, 即第二类 cool number 的位数不超过 80。

第一类 cool number 只需要扫描一遍即可得到。对于第二类 cool number，可以发现这类数并不多，通过搜索发现只有几万个，每次询问在集合中二分就能得到答案。由于搜索的复杂度较高，而在数位和较大的时候，第二类 cool number 较少，可以打出一部分的表。

时间复杂度 $O(\log n + \log |S|)$，S 为第二类 cool number 的集合。

JUNE12.CLOSEST

题目大意

给出两组点，对于 A 组中的点求出 B 组中离它最近的点。

数据规模：$1 \le N \le 50000$。

算法讨论

暴力 K-D tree 就可以了。

时间复杂度 $O(N\sqrt{N})$。

JULY12.DGCD

题目大意

在 n 个节点的树上做 m 次操作，存在两种操作：询问树上两点间路径上权值的最大公约数；树上两点间路径上权值加上 c。

数据规模：$1 \le N \le 50000$，$1 \le m \le 50000$，$0 \le c \le 10^4$。

算法讨论

路径加权值可以用树剖来维护。如果没有区间修改可以直接维护 gcd，考虑把区间修改转化成单点修改。根据 $gcd(a_1, a_2, \ldots, a_k) = gcd(a_1, a_2 - a_1, \ldots, a_k - a_{k-1}$，可以直接把序列差分，差分后就可以用单点修改来表示之前的区间修改。于是用树剖就能解决。

时间复杂度 $O(m \log^3 n)$。

JULY12.EST

题目大意

在 n 个节点的树上做 m 次操作，存在两种操作：询问树上两点间路径上权值的最大公约数；树上两点间路径上权值加上 c。

数据规模：$1 \le N \le 50000$，$1 \le m \le 50000$，$0 \le c \le 10^4$。

算法讨论

路径加权值可以用树剖来维护。如果没有区间修改可以直接维护 gcd，考虑把区间修改转化成单点修改。根据 $gcd(a_1, a_2, \ldots, a_k) = gcd(a_1, a_2 - a_1, \ldots, a_k - a_{k-1}$，可以直接把序列差分，差分后就可以用单点修改来表示之前的区间修改。于是用树剖就能解决。

时间复杂度 $O(m \log^3 n)$。

AUG12.MAGIC

题目大意

两个魔术师在一个有 N 个房间 M 条双向路的地方玩一个游戏。

在游戏的一开始，第一个魔术师在 1 号房间，第二个在 2 号房间，每个魔术师都有 p 点法力。每个魔术师在自己的回合有三个步骤要做。假设当前回合的魔术师为 A 魔术师。

第一步，A 魔术师可以从当前房间沿着已有的路一直走 (没有步数限制)，最终停在某个房间并宣告这一步骤结束。如果他停的房间和另一个魔术师恰好相同，那么游戏结束，A 魔术师胜利，否则继续。

第二步，A 魔术师必须增加一条边原来没有的路。如果不能做到那么 A 魔术师失败，否则继续。

第三步，如果 A 魔术师的法力值为正数，他可以选择消耗 1 点法力值，传送到任何一个房间。当然他也可以什么也不做。

你已经知道了这个地方房间和路的所有信息以及魔术师的法力值。你需要计算出哪个魔术师会赢，如果他们都用的是最优策略。

AUG12.MAGIC continued on next page...

数据规模：$1 \le T \le 100, 2 \le N \le 7777, 0 \le M \le 10000, 0 \le P \le 10000$。

算法讨论

首先考虑 N 是奇数的情况，可以发现局面的最终状态一定是两个人分别在两个完全子图中且没有其余的点。那么可知回合数为 $\frac{N*(N-1)}{2} - Size_1 * Size_2 - M$，判断奇偶性就可以知道获胜方。当 N 是偶数的时候，先考虑 $p = 0$ 的情况，此时两人无法移动位置，且两人期望的最终状态不同。由于奇数连通块的块数是偶数，结果只与初始块的大小有关。对于 $p > 0$ 的情况，初始块的大小已经没有意义，结果只与奇数连通块个数和偶数连通块个数有关，分情况讨论。

时间复杂度 $O(N + M)$。

AUG12.GTHRONES

题目大意

A 和 B 在玩一个数字游戏。规则如下：1. 一开始，有 N 个数字写在一张纸上。2. 双方轮流，A 先手。3. 在第一轮中，A 先选择一个纸上的数字，把它称做游戏的当前数字。4. 之后从第二轮开始执行当前回合的人都按如下操作：我们称现在的当前数字为 u。将 u 从纸上擦去 (如果有多个擦去任意一个)。然后选择另一个纸上的数字 v 作为当前数字，v 要满足与 u 刚好相差一个质因子。5. 无法完成操作的人输。计算如果两个人都采用最优策略谁会获胜。

数据规模：$1 \le N \le 500$，$1 \le u \le 10^{18}$。

算法讨论

边两边的节点质因数个数差 1，所以这个图是二分图。在二分图最大匹配中的点即为合法点，由于相同的数很多，可以转化为最大流。最后从起点和终点 bfs 一遍，遍历到的点即为合法点。

时间复杂度 $O(N^3)$。

SEPT12.KNGHTMOV

题目大意

有一个无限大的棋盘，起初有一个骑士在 $(0, 0)$ 点。若骑士的当前位置是 (x, y)，下一次移动可以到 $(x + A_x, y + A_y)$ 或 $(x + B_x, y + B_y)$。在棋盘上有 K 个格子是无法到达的，问有多少种方案使骑士到达 (X, Y) 点。

数据规模：$1 \le K \le 15, 1 \le A_x, A_y, B_x, B_y, X, Y \le 500$。

算法讨论

如果向量 (A_x, A_y) 和 (B_x, B_y) 线性无关，那么转换坐标后就是每次移动一格的距离，考虑到有一些格子无法到达，用容斥解决。

如果线性相关，问题转化成了一维的问题，坐标范围只有 250000，前后扫两遍转移即可。

时间复杂度 $O(2^K K^2 + L^2)$。

SEPT12.SIMNIM

题目大意

给出 N 个数，要把这些数分成尽量多堆，使得异或和是 0 的堆数最多。

数据规模：$1 \le N \le 1000$。

算法讨论

把 N 个数重排，每次找出一组异或和是 0 的数，找不出就结束。多次进行这个操作取最优解。

时间复杂度 $O(60TN)$，T 表示随机次数。

OCT12.MAXCIR

题目大意

给出一个三角形 ABC 和 N 个操作，每个操作有两个参数 x_i、y_i。使用第 i 个操作可以让点 A 的 x 坐标增加 x_i，y 坐标增加 y_i。

现在需要从这 N 个操作中取出至多 K 个，来让三角形的周长最大。

数据规模：$K \leq N \leq 500$，需要答案的绝对误差必须小于 10^{-12}。

算法讨论

设点 A 的最终坐标为 (X,Y)，可以证明存在一组向量 (u,v)，在最大化 $f(X,Y) = uX + vY$ 的同时三角形的周长最大。那么枚举向量 (u,v)，同时维护所有操作的大小关系就可以了。需要枚举的关键向量包括两个操作值相同时的向量和某个操作改变正负时的向量。

由于本题的精度要求很高，在排序的时候要用叉积判断大小，最后计算距离要用到开根操作，通过以下方式来计算。

令 $sqrt(S) = I + D$，其中 I 为整数部分，D 为小数部分。I 可以直接得到。

$$I^2 + 2ID + D^2 = S$$
$$D = \frac{S - I^2}{I + \sqrt{S}}$$

时间复杂度 $O(N^2 \log N)$。

OCT12.MAXRECT

题目大意

给出一个 $H \times W$ 的整数矩阵 A，行标号 0 至 H−1，列标号 0 至 W−1，求一个子矩阵使其中元素之和尽可能大。

这个子矩阵不要求是连续的，即求出一些行和一些列，选取这些行列相交处的元素，输出这些行列。

数据规模：$200 \leq H, W \leq 300$。

算法讨论

首先全选，之后按照每次翻转一行或一列模拟退火。

时间复杂度 $O(TH^2)$，T 表示模拟退火次数。

NOV12.COUNTARI

题目大意

给定 N 个整数 $A_1, A_2, \cdots, A_N$, 求有多少种选择得他们构成一个等差数列。

数据规模：$1 \leq N \leq 100000, 1 \leq A_i \leq 30000$。

算法讨论

可以方便得到一个复杂度为 $O(N * A_i)$ 的算法：枚举每个中间数，再枚举左边数的大小，这样可以得到右边数的大小，累加即为答案。其中第二个枚举可以用八指针优化。这题也可以用分块 +fft 解决，但常数很大。

时间复杂度 $O(N * A_i / \alpha)$。

NOV12.MARTARTS

题目大意

两个 N 人的队伍要打 N 场比赛，每人都要出场一次，第一个队伍的第 i 人和第二个队伍的第 j 人比赛的得分是 $A_{i,j}:B_{i,j}$。现在队伍使得自己队伍的分数减去另一个队伍的分数最小，在相同的情况下让自己队伍的分数最多。第一个队伍可以安排两队的出场顺序，第二个队伍可以让一场比赛无效。

数据规模：$1 \le N \le 100$。

算法讨论

如果第一个队选出 $P_1:Q_1, P_2:Q_2, \cdots, P_N:Q_N$，第二个队删去的一定是 $P_i - Q_i$ 最小的一个。按照 $A_{i,j} - B_{i,j}$ 从小到大的顺序加入边，问题就转化为使用现有的边做二分图最大权匹配。这用 KM 的想法，每加入一条边，从端点开始找增广路并更新顶标。在差相同的情况下要让自己的分数最大，可以把边权设为二元组。

时间复杂度 $O(N^4)$。

DEC12.DIFTRIP

题目大意

给出一棵 N 个点的树，每个节点的权值是它的度数，求有多少条不同的路径。

数据规模：$1 \le N \le 10^5$。

算法讨论

建出后缀自动机，枚举每个节点求和即为答案。字符集很大，需要用 map 来存。

时间复杂度：$O(N \log N)$。

JAN13.ANDOOR

题目大意

求 n 个圆的并在矩形范围内的周长。

数据规模：$1 \le n \le 1000$。

算法讨论

分别对每个圆算出贡献，即不被其他圆包含部分的长度。用余弦定理求出其他圆在所求圆上对应的角度范围，排序有计算出不被包含的角度和，就可以算出长度。累加即为答案。

时间复杂度 $O(n^2 \log n)$。

JAN13.CUCUMBER

题目大意

给定 B 个 $n*n$ 的矩阵 A_i，对于数对 $(x,y)(x<y)$，定义矩阵 B，满足 $B_{i,j} = \sum_{k=1}^{n} A_{x,i,k} * A_{y,j,k}$。一个排列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) 是好的当且仅当排列 (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。

数据规模：$1 \le n \le 60$ $1 \le B \le 8000$。

算法讨论

定义矩阵 C 满足 $C_{i,j} = (B_{i,j}+1) \ mod\ 2$，数对 (a,b) 是好的等价与 $det(C)$ 是奇数。在每个矩阵 A 最后后补上全为 1 的一行，令 $A_{i,j}$ 为矩阵 A_i 删掉第 j 行后的矩阵，那么有 $det(C) = \sum_{i=1}^{n+1}(det(A_{a,i}) + det(A_{b,i}))$，求出所有的 $det(A_{a,i})$ 后就能得到答案。如果 A_i 不满秩，那么 $det(A_{i,j})$ 一定为 0，否则可以把 A 消元成 $n*n$ 的单位矩阵加一行的矩阵 T。设多出的一行为第 k 行，那么

当 $j<k$ 时，有 $det(A_{i,j}) = D_{j,k}$。

当 $j=k$ 时，有 $det(A_{i,j}) = 1$。

当 $j>k$ 时，有 $det(A_{i,j}) = 0$。

时间复杂度 $O(B^2 + n^2 * B)$。

FEB13.ROC

题目大意

有一个内角都为 $90°$ 或 $270°$ 的多边形房间。每一个 $90°$ 的内角上都站着一个小朋友，相邻的两个小朋友可以交换位置，速度是每秒一格。对于每个询问 (i, j)，求第 i 个小朋友和第 j 个小朋友的最短相遇时间。

数据规模：$Q \leq 10000$，$n, m \leq 2500$。

算法讨论

小朋友之间构成了一个环，如果得到了这个环的信息，对于每组询问只需要在两个人之间二分出相遇位置就可以了。通过让一个小朋友环绕房间一周来得到这个环，以逆时针为例。让小朋友始终保持右手边是墙，当前方是墙时左转，当右方没墙时左转。需要注意判断是否回到初始位置时还要方向正确，判断是否经过一个小朋友时要保证右边是墙。

时间复杂度 $O(nm + Q \log P)$，P 为小朋友数。

FEB13.QUERY

题目大意

在 N 个点的带权数上做 M 个操作：

c x y a b ：节点 x 到节点 y 的路径上加首项 a 公差 b 等差数列。q x y ：询问节点 x 到节点 y 的点权和。l x ：返回第 x 次修改后的状态。

需要在线算法

数据规模：$1 \leq N, M \leq 10^5$。

算法讨论

树剖之后问题变为区间操作。由于两个等差数列相加后仍为等差数列，故标记具有可加性，可以用线段树来维护区间。至于可持久化操作只需要主席树就可以了。

时间复杂度 $O(n \log^2 n)$

MARCH13.LECOINS

题目大意

有 n 种硬币，第 i 种硬币的面值为 V_i，颜色为 C_i，且有无穷个。有 Q 组询问，从这些硬币中取一些使得他们的和是 S，求最小颜色数。

数据规模：$1 \leq n \leq 30$，$1 \leq V_i$，$Q \leq 2 \times 10^5$。

算法讨论

先去掉任意一种面值为 v 颜色为 c 的硬币。用 $dp_{i,j}$ 表示使用了 i 种颜色、面值和模 v 为 j 的最小面值。可以用类似背包的方式转移，需要特殊处理颜色为 c 的硬币。对于一个询问 S，比较它与 $dp_{i,S\%v}$ 的大小就可以判断是否存在一组解。

时间复杂度 $O(n^2V)$。

MAY13.QTREE

题目大意

在一棵环加外向树中支持两个操作：

1. 对 u 到 v 的最短路径上的所有边的权值取相反数。

2. 在 u 到 v 的最短路径上，找出一个连续的边的集合，使得集合中边的权值之和最大。

数据规模：$1 \leq n, m \leq 10^5$。

MAY13.QTREE continued on next page...

算法讨论

该问题的序列操作是经典的线段树问题。

如果图是一棵树，可以用树剖转化为序列操作，对于环加外向树而言，只需要再用一棵线段树来维护环上的边。

时间复杂度 $O(n\log^2 n)$。

JUNE13.TKCONVEX

题目大意

给出 n 条线段，询问是否可以取出 $2k$ 条线段组成两个 k 条边的凸包。

数据规模：$2k \le n \le 1000$，$3 \le k \le 10$。

算法讨论

k 条线段 $a_1, a_2, \ldots, a_k$，$a_1 \le a_2 \le \ldots \le a_k$ 能组成一个凸包需要满足 $a_1 + a_2 + \ldots + a_{k-1} > a_k$

将 n 条线段排序，枚举两个凸包中的最长边 l_{B_1}，l_{B_2}。

若 $B_2 - B_1 \ge k$，那么判断 $l_{B_1}, l_{B_1-1}, \ldots, l_{B_1-k+1}$ 能否组成凸包，$l_{B_2}, l_{B_2-1}, lB_2 - k + 1$ 能否组成凸包。

若 $B_2 - B_1 < k$，那么 $l_{B_1+1}, l_{B_1+2}, \ldots, l_{B_2-1}$ 属于凸包二，暴力枚举 $l_{B_2-2k+1}, l_{B_2-2k+2}, \ldots, l_{B_1-1}$ 属于那么凸包，判断是否存在一组解。

时间复杂度 $O(nk\binom{k}{2k})$。

JUNE13.SPMATRIX

题目大意

称一个 $n * n$ 的整数矩阵是好的当且仅当它满足：

1.$A_{i,i} = 0$。

2.$1 \le A_{i,j} < n - 1 (i \ne j)$。

3. 对于每一个整数 $0 \le k \le n - 2$，矩阵中至少有一个位置的值是 k。

4.$A_{i,j} \le \max(A_{i,k}, A_{k,j}) (1 \le i, j, k \le n)$

给出 m 组 n，求 $n * n$ 的好矩阵个数。

数据规模：$3 \le n \le 10^7$，$1 \le m \le 10^5$。

算法讨论

暴力计算前几项，oeis 后可以得到答案为 $\frac{n!(n-1)!}{3 \times 2^n}(3n - 4 - 2H_{n-1})$，预处理后 $O(1)$ 回答询问。需要卡常。

时间复杂度 $O(n + m)$。

JUNE13.CHAORNOT

题目大意

比赛开始前一个礼拜,Chef 想出了一个有趣的脑洞,出了下面这道题：给你 N 个整数 $A_1, A_2, \cdots, A_N$，如果里面存在三个数字能构成等差数列，就打印 YES 或 NO。所有整数在 0 到 9999 之间，且不同。正如 Chef 预料，比赛途中，很多队伍没能想出一个高效的算法，但是比赛的最后一分钟，突然多了一个 10 行代码的提交，而且是正确的。这个简短的解答时这样做的：如果 $N >= 2000$ 打 YES，否则暴力。Chef 感觉身败名裂，他（她）坚信这个解答是错的，现在他想叉掉这个解，请你帮忙。

数据规模：$N, B_i \le 10^5$。

算法讨论

多次随机排列，如果出现不合法就退出。

JUNE13.CHAORNOT continued on next page...

时间复杂度 $O(TN)$。

JULY13.RIVPILE

题目大意

给定 n 个点和 m 种半径的圆, 每种圆都有一个价格。现在你需要以每个点位圆心选择一种圆, 使得存在从直线 y=d 出发经过圆上的点的路径到达 y=0 的路径，求最小费用。数据规模：$1 \le n, m \le 250$。

算法讨论

总共有 $n * m$ 个不同的圆。建成 $n * m$ 个点，两个点之间有边当且仅当两个圆有公共部分。如果暴力连边就会有 n^2m^2 级别的边数，需要优化。考虑两个有公共部分的圆，如果圆的半径变大，它们依然有公共部分。那么只需要连两圆恰好有公共部分的边和相同圆心点之间的边。边数变为 n^2m。

时间复杂度 $O(nm\log(n^2m))$。

AUG13.LYRC

题目大意

给出 N 个单词和 M 条句子，求每个单词出现的次数。数据规模：$1 \le N \le 500, 1 \le M \le 100$。

算法讨论

先把单词建成 AC 自动机，每条句子在自动机上完成转移。最后统计每个单词出现的次数。

时间复杂度 $O(N * Len_1 + M * Len_2)$。

AUG13.PRIMEDST

题目大意

给出一棵 N 个点的树，求在树上等概率随机两个不同的点，距离是质数的概率。

数据规模：$1 \le N \le 50000$。

算法讨论

可以使用点分治，在合并两个子树信息的时候发现是个卷积，用 fft 来加速运算。

时间复杂度 $O(Nlog^3(N))$。

AUG13.DELNMS

题目大意

给出一个数组 A_i，每次删除操作可以删去一些数，需要满足这些数大小相同，且位置呈等差数列。

数据规模：$1 \le N \le 100000$。

算法讨论

对于出现次数较小的数可以先处理掉，之后把数组拆成一些块，每一块中贪心或 dp 解决。可以多次改变块的大小。

时间复杂度 $O(TN)$。

SEP13.TWOROADS

题目大意

在平面上有 N 个点，需要找出两条直线，使得所有点到直线的距离平方和最小。

数据规模：$1 \le N \le 100$。

SEP13.TWOROADS continued on next page...

算法讨论

考虑两条相交直线，它们的控制范围可以用两条互相垂直的分割线划分。那么可以枚举其中一条分割线，再枚举与这条线垂直的线，这样把点分成了两部分。问题转化为求一条直线，使得集合中的点到这条直线的距离平方和最小。通过数学推导可以得到答案。

时间复杂度 $O(N^3 \log N)$。

OCT13.FN

题目大意

定义 F_n：

$F_n = n \ (n \leq 1)$

$F_n = F_{n-1} + F_{n-2} \ (n > 1)$

给出 M,C，求满足 $F_n \ mod \ M = C$ 的最小 n。

数据规模：$1 \leq T \leq 100$ $0 \leq C < M \leq 2 * 10^9$，$M$ 是质数且 $M \ mod \ 10$ 是完全平方数。

算法讨论

斐波那契的通项公式是

$$F_n = \frac{1}{\sqrt{5}}(\frac{1+\sqrt{5}}{2}^n - \frac{1-\sqrt{5}}{2}^n)$$

设 $x = \frac{1+\sqrt{5}}{2}$

需要满足

$$x^n - (-x)^{-n} = \sqrt{5} * C$$

以 $n \ mod \ 2 = 0$ 为例，令 $a = \sqrt{5} * C$ 可以解得

$$x^n = \frac{a \pm \sqrt{a^2 - 4}}{2}$$

再使用 BSGS 就可以解出 x 的值。

时间复杂度 $O(\sqrt{N})$。

NOV13.MONOPLOY

题目大意

给定一棵 N 个点的有根树，初始状态下点的权值各不相同。需要进行 M 次操作：1. 把 x 到根的路径上的点权值改为一个新的权值。2. 询问 x 子树中的点到根路径上不同权值数的平均值。

数据规模：$1 \leq n, m \leq 10^5$。

算法讨论

修改操作相当于 lct 中的 Access 操作，而 Access 时改变边的虚实性就会影响到子树中的点，只要用树状数组来支持子树加减就可以了。

时间复杂度 $O(M \log^2 N)$。

NOV13.QPOINT

题目大意

有 N 个不相交的多边形，点数之和为 K。每次询问一个点在哪个多边形内。强制在线。

数据规模：$1 \leq N, Q \leq 10^5$，$1 \leq K \leq 3 * 10^5$。

算法讨论

NOV13.QPOINT continued on next page...

先考虑离线做法。判断一个点在哪个多边形中可以从这个点垂直向下做一条射线，计算这条射线经过了几条边，如果是奇数表示这个点在多边形中，否则表示这个点在多边形外，同时这个点下方的那条边所在多边形就是这个点所在的多边形。那么用扫描线来处理问题，需要维护当前 X 坐标下的边的上下关系，考虑到边的位置始终不会改变，建一棵平衡树来维护边的关系，需要用到的操作包括插入一条边、删除一条边、询问该坐标下某一个 Y 值下方的第一条边。

当原问题需要在线算法。这可以通过扩展离线算法得到，即把平衡树可持久化。使用非旋转机制的平衡树就可以完成平衡树的可持久化。

时间复杂度 $O(M \log N)$。

DEC13.QTREE6

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)。

数据规模：$1 \le n, m \le 10^5$。

算法讨论

把节点拆成三个点 (s, b, w)，s 无权，b、w 有权。s 与 b 相连表示该点为黑色，与 w 相连表示该点为白色。同时若该点为黑色，则 s 与父亲的 b 相连，白色同理。由此可以发现 1 操作只需要计算该节点的 s 所在连通块的权值和即可。操作 0 需要改变的边只有 4 条。用 lct 维护。

时间复杂度 $O(mlogn)$。

DEC13.REALSET

题目大意

给定一个长度为 N 的整数数组 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 $\sum_{i=0}^{n-1} A_i \times B_{(i,j)\ mod\ n} = 0$。

数据规模：$1 \le T \le 100, 1 \le N \le 3 \times 10^4$。

算法讨论

定义矩阵 M，满足 $M_{i,j} = A_{(i+j)\ mod\ n}$，问题等价于判断矩阵 M 是否满秩。如果 $\gcd(f(x), x^n - 1)$ 的次数是 d，那么矩阵的秩就是 $n - d$，其中 $f(x) = \sum_{i=0}^{n-1} A_i x^i$。问题等价于判断 $x^d - 1 \mid f(x) \times \prod_{p|d} (x^{d/p} - 1)$，逐个判断即可。

时间复杂度 $O(n \log n)$。

JAN14.CNTDSETS

题目大意

在 N 维中间中找出一个点集，使得点的每一维都是在 $[0, D]$ 间的整数，并且至少存在一个点，它的某一维是 D；每一维都存在至少一个点，它的这一维是 0。

数据规模：$1 \le N \le 1000$，$1 \le D \le 10^9$。

算法讨论

考虑容斥，可以得到下面这个公式：

$$ans = \sum_{i=0}^{N} (2^{D^i*(D+1)^{n-i}} - 2^{(D-1)^i*D^{n-i}} * \binom{i}{n} * (-1)^i$$

直接计算即可。

时间复杂度 $O(N \log Mod)$。

JAN14.TAPAIR

题目大意

求一个 N 个点 M 条边的图中存在多少个无序二元组 (u, v)，使得删掉 u 和 v 这两条边后图不连通。

数据规模：$1 \le N \le 100000$，$1 \le M \le 300000$

算法讨论

先建出 dfs 树，分情况讨论，满足条件的二元组有如下三种情况：

1. 桥边 + 任意边，这个只需要求出桥边个数就可以了。

2. 树边 + 非树边，删去树边后把 dfs 树分成了两个部分，只有仅存在一条连接这两部分的边时才存在一组解。

3. 树边 + 树边，这两条树边都不是桥边。每条树边记跨过这条边的非树边集合。删去两条边后 dfs 树被分成了三个部分，如果两条边的集合相同，那么非树边只能连接两端的部分，图不连通。否则即连通。判断集合是否相同可以用哈希，对每条非树边分配一个随机值，集合用元素的异或和来表示。

时间复杂度 $O(M \log M)$。

FEB14.DAGCH

题目大意

给出一个 N 个点 M 条边有向图，对每个点，求半必经点是这个点的点数。

数据规模：$1 \le N \le 100000$，$1 \le m \le 200000$。

算法讨论

套用 dominator tree 来计算每个点的半必经点。

时间复杂度 $O(n)$。

FEB14.COT5

题目大意

需要维护一个 treap，支持三个操作：

1. 加入一个 key 是 k，weight 是 w 的新节点。

2. 删除一个 key 是 k 的节点。

3. 求 key 为 ku,kv 的两个点之间的距离。

保证任意两点 key 和 weight 都不同。

数据规模：$1 \le N \le 200000$。

算法讨论

将 treap 中序遍历后就可以把问题转化为序列的情况。

首先把所有操作离散化，加入和删除的点都可以知道它在序列中的位置。考虑询问，要求距离就需要求出两个点和它们 lca 的深度。容易发现 lca 就是这两个点之间 weight 最大的点。接下来就要求某一点的深度，让这个点向序列的某一端走，如果遇到一个权值比它大的就把这个点当做父亲，这样走到端点就可以计算出深度。

那么用线段树来维护这个计算过程，需要解决合并区间和询问。对于区间 u，它有两个子区间 l,r。询问的形式为从 u 的左端以 w 的权值出发，到 u 的右端经过的长度。这可以先询问 l 的最大值 maxl,

如果 w>maxl，直接跳过左区间，如果 w<maxl，走过左区间后的权值就是左区间的最大值，需要维护以初始权值 0 走完区间的长度，合并区间信息的时候需要套用询问。

时间复杂度 $O(n\log^2 n)$。

MARCH14.GERALD07

题目大意

给定一个无向图和 Q 组询问 L_i,R_i，对于每组询问，求保留编号在 $[L_i, R_i]$ 范围内的边时的连通块个数。

数据规模：$1 \leq N, M, Q \leq 200000$。

算法讨论

离线处理询问。对于枚举到的当前右端点 r，需要求出编号在 $[1, r]$ 范围内的边所组成的最大生成森林。若当前询问为 $[l, r]$，那么这个最大生成森林中每存在一条 $[1, l-1]$ 范围内的点就意味着增加了一个连通块。

考虑如何维护最大生成森林。对于每次加边，如果产生了一个环，就要去掉这个环中编号最小的一条边，这可以用 lct 来进行维护。再使用一个树状数组来记录当前在最大生成森林中的边。

时间复杂度 $O(n\log n)$。

MARCH14.STREETTA

题目大意

你需要维护以下操作：

1 u v a b：新增纪念品操作，表示对于编号为 u 的商品，新增一种价格为 b 的商品，对于编号为 $u+1$ 的商店，新增一种价格为 $b+a$ 商品。更准确的说，对于编号为 $u+i$ 的商店，新增一种价格为 $b+i{\times}a$ 价格的商品（$0 \leq i \leq v-u$）。

2 u v a b：税费调整操作，表示对于编号为 u 的商店，税费增加 b，对于编号为 $u+1$ 的商店，税费增加 $b+a$。更准确的说，对于编号为 $u+i$ 的商店，税费增加 $b+i{\times}a$（$0 \leq i \leq v-u$）。

3 i：询问操作，询问顾客在编号为 i 的商店，购买一件纪念品最高的花费是多少。顾客的花费为这间商店中价格最高的纪念品的价格，加上累计对这间商店产生影响的税费的总和。

算法讨论

修改操作加入的都是等差数列，可以把这看作线段。使用线段树来维护，求和操作标记具有可加性能直接维护。求最大值操作时通过比较新加入的线段和当前线段的关系，根据在左端点、右端点、中点的大小关系来决定向左儿子或右儿子转移。

APRIL14.GERALD08

题目大意

给一棵 n 个节点的树，树边的颜色为红色或蓝色。有两个玩家 A,B，两人轮流操作，A 每次选择一条红色边删除，B 每次选择一条蓝色边删除。每次删除后和根不连通的点会被删除。如果无法操作就算输。询问 A 先手、B 先手的情况下谁会取胜。

数据规模：$1 \leq N \leq 100000$。

算法讨论

每个节点定义一个局面函数 f。叶节点的函数值为 0，对于一个非叶节点，计算它的每个儿子节点对它的贡献：

1. 红边儿子 i：令 s 为满足 $f_i + s > 1$ 的最小正整数，贡献为 $\frac{f_i+a}{2^{a-1}}$。
2. 蓝边儿子 i：令 s 为满足 $f_i - a < -1$ 的最小正整数，贡献为 $\frac{f_i-a}{2^{a-1}}$。

判断获胜情况只需要讨论根节点的函数值，如果函数值为 0，那么后手赢，如果是整数，那么 A 赢，否则 B 赢。

这个结论可以通过 $SurrealNumber$ 得到。

由于涉及的数很大，需要使用高精度，在求和时可以用类似启发式合并的方法。

时间复杂度 $O(n\log^2 n)$。

MAY14.ANUDTQ

题目大意

给出一棵 N 个节点的点权树，支持 4 种操作：

1. 加入一个叶节点。
2. 删除子树。
3. 子树中所有节点加上某个值。
4. 询问子树权值和。

强制在线。

数据规模：$1 \leq n, m \leq 10^5$。

算法讨论

需要改变连通情况，考虑用 LCT 来维护。但 LCT 并不能很好地解决子树操作，需要进行转化。用 $size_i$ 表示子树大小,$value_i$ 表示这个点的子树需要增加的值。节点 i 的子树权值和就是 $size_i \sum_j value_j + \sum_k value_k \times size_k$。那么每次修改变为修改一条路径上的 $size, \sum_k value_k \times size_k$ 值，询问变询问路径上的权值，可以用 LCT 来维护。

时间复杂度 $O(n\log n)$。

MAY14.SEINC

题目大意

给出两个长为 N 的数组 A,B，每次选出 A 数组的一个区间 $[l, r]$ 加 1，询问最少进行多少次操作能使数组 A 变成 B。

数据规模：$1 \leq N \leq 10^5, 0 \leq A_i, B_i \leq 3$。

算法讨论

先求出 A,B 数组的差 C，在令 $D_i = C_{i+1} - C_i$。那么每次选出 A 数组的一个区间，就相当于将 D 数组的两个位置改变权值，最终目标是 D 数组转化为全 0 数组。

考虑使用贪心，从左往右存下剩余的-1,-2,-3 数量，每次遇到 1,2,3，都进行合并。

时间复杂度 $O(n)$。

JUNE14.TWOCOMP

题目大意

给定一棵 N 个点的树和两个大小为 M 的集合，集合中的元素是树上的链，要分别找出两个集合的子集，并保证不同集合的链不相交，要求权值和最大。

数据规模：$1 \leq N \leq 10^5$，$1 \leq M \leq 700$。

算法讨论

可以看出该问题是个最小割。源向 A 集合中的点连容量为该点权值的边，B 集合中的点向汇连容量为该点权值的边。若 A 集合中的某条链和 B 集合中的某条链相交，在这两个点之间连容量为无穷大的边。判断两条链是否相交可以算出 lca，拆成两 (一) 条链后分别判断。

时间复杂度 $O(M\log N + Flow)$。

JUNE14.SEAARC

题目大意

N 个点排成一行，第 i 个点的颜色是 A_i，如果两个点颜色相同，在这两个点之间连一条同色圆弧，求不同颜色圆弧交点数量。

数据规模：$1 \le N, A_i \le 10^5$。

算法讨论

两条圆弧之间的关系有三种 aabb,abab,baab，只有 abab 的情况会产生交点，可以通过总数减去 aabb,baab 的数量来计算答案。aabb 可以枚举其中一条边的位置来统计。baab 需要分类讨论，存在两种情况：两种颜色有一种个数大于 $\sqrt{\frac{n}{\log n}}$；两种颜色个数都不大于 $\sqrt{\frac{n}{\log n}}$。这都可以用预处理加数点的方式计算。另外这道题可用用一些复杂度不正常的分块算法调整块大小通过。

时间复杂度 $O(n\sqrt{n\log n})$。

JULY14.GNUM

题目大意

称两个长度为 N 的数组 A,B 相似，当且仅当对于所有 i，满足 $C(A, A_i) = C(B, B_i)$。其中 $C(X, x)$ 等于满足 $X[j] < x$ 的 j 的数目。对于两个排列 P_1,P_2，定义函数 $f(P_1, P_2)$ 等于满足 $P_1[l, r]$ 相似于 $P_2[l, r]$ 并且 $P_1[l, r]$ 包含不超过 E 个逆序对的数对 (l, r) 的数目。对 P_1,P_2 取遍所有 N 个元素的排列 $f(P_1, P_2)$ 的总和是多少。

数据规模：$1 \le T \le 500, 1 \le N \le 500, 1 \le E \le 10^6$。

算法讨论

首先可以用简单的 dp 预处理出长度为 i 的排列中逆序对数不超过 j 个排列个数。接着枚举子串的长度 l，计算这种长度的子串的贡献为 $(n-l+1)(n-l)!\binom{n}{l} \times dp_{l,E}$。

时间复杂度 $O(tn + n^3)$。

JULY14.SEAEQ

题目大意

两个长度为 N 的数组 A,B 是相似的当且仅当对于任意的 i 都有

数据规模：$1 \le N \le 400$。

算法讨论

把数对当做点，满足条件的两条边之间连边，根据数对两个数的大小关系可以发现这是个二分图，最大匹配数就是答案。但这样的边数太多，需要优化。对于每个质数新建一个点，在 gcd 是这个质数倍数的点和这个点之间连边，求最大流就是答案。

时间复杂度 $O(flow(n^2, n^2 \log n)$。

AUG14.SIGFIB

题目大意

给出 N,M，求

$$\sum 6xyzfib_xfib_yfib_z[x+y+z=n] \; mod \; m$$

数据规模：$1 \le N \le 10^18$，$1 \le M \le 10^5$。

算法讨论

求出这个数列的母函数

$$f(x) = 6x^3(x^2+1)^3g(x)$$

其中

$$[x^n]g(x) = \frac{1}{3125}\left[\binom{10}{5}fib_{n+1} + \binom{9}{4}(n+1)(fib_{n+2}+2fib_{n+1}) + 5\binom{8}{3}\binom{n+2}{2}fib_{n+3} + 5\binom{7}{2}\binom{n+3}{3}(fibn+4$$

计算出各项的和即可。时间复杂度 $O(\log N)$。

AUG14.PUSHFLOW

题目大意

给出 N 个点 M 条边的无向连通图，满足每个点至多属于一条边。支持两种操作：

1. 改变某条边的权值。

2. 询问以 S 为源，T 为汇的最大流。

算法讨论

最大流等于最小割，需要割去一些边使得 S、T 不相连。如果 S、T 在同一个环内，在两边都删去最小权值边。如果在不同环内，先把环缩成点，构成了一棵树，可以选择割去一条边或割去一个点，割去一个点意味着删去两点之间两边的最小权值边。这些可以通过熟练剖分维护，在重链时暴力向上跳，在轻边时重新计算答案。

时间复杂度 $O(N+M+q\log^2 N)$。

SEPT14.QRECT

题目大意

在二维平面中支持三种操作：

1. 插入一个矩形。

2. 删除一个矩形。

3. 给定一个矩形，询问有多少个矩形和这个矩形相交。

数据规模：$1 \le Q \le 10^5$。

算法讨论

将与给定矩形相交的矩形所在位置分情况讨论。需要维护某个区间和某个矩形范围中的点数和，那么用线段树和树套树来维护信息。

时间复杂度 $O(Q\log Q)$。

SEPT14.FIBTREE

题目大意

给一棵 N 个点的树，支持四种操作：1. 将 u 到 v 路径上的第 i 个点加 fib_i，fib_i 表示斐波那契数列的第 i 项。

2. 询问以 x 为根时 y 的子树和。

3. 询问 u 到 v 路径上节点的权值和。

4. 返回第 i 次操作后的状态。

数据规模：$1 \le N, Q \le 10^5$。

SEPT14.FIBTREE continued on next page...

算法讨论

由于要处理路径上的问题，考虑用树链剖分来维护。

处理第一个操作，先写出斐波那契数列的通项：

$$fib_i = \frac{1}{\sqrt{5}}[(\frac{1+\sqrt{5}}{2})^i - (\frac{1-\sqrt{5}}{2})^i]$$

用 5 在模 10^9+9 下的二次剩余来代替 $\sqrt{5}$，那么就相当于加上了两个等比数列，而区间加等比数列可以用线段树加标记维护。

对于操作二，通过判断 x 与 y 的位置关系来确定子树包括的范围。对于操作四，把线段树可持久化即可。

时间复杂度 $O(Q\log^2 N)$。

OCT14.TRIPS

题目大意

给一棵 N 个点的树，边的权值是 1 或 2，有 Q 组询问，每组询问 u_i,v_i,d_i，表示要从 u_i 走到 v_i，每天最多经过 d_i 的长度，且每天结束都要停留在某个点上，求最小要走几天。

数据规模：$1 \leq N, Q \leq 10^5$。

算法讨论

令 L 为 $lca(u,v)$，从 u 到 v 的答案等于从 u 到 L 的答案加从 v 到 L 的答案加剩余部分所需步数。那么模拟从一个点走到它的某个祖先节点就可以算得答案，发现权值只有两种，可以压位来使得一次跳 10 格。需要预处理当前还有 i 长度可走，上方几条边的状态 j 时需要走几天，还剩多少长度可走。

时间复杂度 $O(NQ/10)$。

OCT14.BTREE

题目大意

给出一棵 N 个节点的树，处理 Q 组询问，每组询问给出 K_i 个数对 (a_i, r_i)，表示距离 a_i 点 r_i 范围内的点都被守护了，求共有多少点被守护了。

数据规模：$1 \leq N, Q, \sum_{K_i} \leq 5 \times 10^5$。

算法讨论

首先用点分治预处理，就可以计算距离某个点某个距离内的点的数量。对于每组询问建出虚树，首先对虚树上的每个点计算出它真正有效的守护区间，如果某个点完全无效可以直接在树上删除。之后使用容斥来去除重复的部分，这只会出现在相邻的两个点上。

时间复杂度 $O(N\log N + \sum K_i \log N)$。

OCT14.CHEFPNT

题目大意

游戏的棋盘是一个 N 行 M 列的矩阵，行从 1 到 N 标号，列从 1 到 M 标号。

初始时，棋盘上有一些黑色的格子，其它的格子都是白色的。

每一步，大厨可以选择任意一个白色的格子和一个方向（左-右或上-下）

如果大厨选择了左-右，那么他会从选定的白色格子开始，向左边依次把格子涂成红色，直到到达了棋盘边界或者到达了某个之前已经被染成了黑色或红色的格子时停止涂色；然后再从选定的格子开始，向右边进行同样的涂色操作。

如果大厨选择了上-下，也与左-右类似，大厨会在选定的格子的上下方向上颜色。

游戏的目标是用尽可能少的步数把所有原来的白色的格子都涂成红色。

数据规模：$1 \leq N, M \leq 100, 1 \leq K \leq min(NM-1, 3000)$。

算法讨论

使用两种方法取最优解。第一种方法是用 $dp[i][j]$ 表示选取 $[1,1]$ 到 $[i,j]$ 的格子填满后所需的最少步数，之后 dp 求解。第二种方法是枚举每一个白色格子，选较长的一种方式染色。

时间复杂度 $O(N^3)$。

NOV14.FNCS

题目大意

给定一个长为 N 的 a 数组和 N 个函数，第 i 个函数 f_i 的值为 $\sum_{k=l_i}^{r_i} a_k$。完成 M 个操作，操作分为两种：

1. 修改 a_i 的值。

2. 询问 $[l,r]$ 区间内函数值的和。

数据规模：$1 \leq N, M \leq 10^5$

算法讨论

将函数序列分块，先考虑维护整块的答案。对于数组 a 中的每个数，通过预处理可以计算出它对每一块的贡献值，那么每 $\sqrt{M}$ 次操作后重新计算所有函数的值，再统计多余的修改操作对每一个块的影响就行了。再考虑非整块的答案，由于这些函数的个数只有 $\sqrt{N}$ 个，可以逐个计算。

要计算某个函数当前的值可以通过维护 a 数组的前缀和数组 b，单点修改转化为区间修改，区间询问转化为单点询问，这样就可以在 $O(1)$ 的时间内计算某个函数的当前值了。

时间复杂度 $O(N\sqrt{N})$。

NOV14.SEAORD

题目大意

有 N 个任务，每个任务需要在第一个机器中跑 A_i 秒，在第二个机器中跑 B_i 秒，并且不能同时跑在两个机器中。现在要给出一个机器的运作顺序使最快结束。

数据规模：$1 \leq N \leq 10^5$

算法讨论

问题的理论下界是 $\min(\sum A_i, \sum B_i, A_i + B_i)$，而这是可以取到的。如果最小值是 $A_i + B_i$，就先把最大的放好，再把其他的加进去，这一定不会出现重复的情况。如果最小值是 $\sum A_i$，可以使用随机，每次随机出一组排列，然后贪心加入任务，判断另一个机器的最晚时间是否更早。

时间复杂度 $O(N * times)$。

DEC14.DIVIDEN

题目大意

使用尺规作图把一个 N 度角 N 等分。

算法讨论

根据定理，在空白平面上尺规作图所能得到的最小的整数角为 $3°$。需要得到的最小角度为 $1°$，那么就要使用 $3°$ 角和 $N°$ 来构造出 $1°$ 角。

分情况讨论：若 $N = 3k$，无解；若 $N = 3k+1$，可以把这个角分成 k 个 $3°$ 角和一个 $1°$ 角；若 $N = 3k+2$，可以把这个角分成 k 个 $3°$ 角和一个 $2°$ 角，把 $2°$ 角平分得到 $1°$ 角。

DEC14.DIVIDEN continued on next page...

考虑得到 3° 角：作等边三角形得到 60° 角，平分得到 30° 角。作直角边长为 1 和 2 的直角三角形得到 $\sqrt{5}$，加 1 后二等分得到 $\frac{\sqrt{5}+1}{2}$。以 $\frac{\sqrt{5}+1}{2}$ 为腰，1 为底作等腰三角形，可得顶角为 36°。减去 30° 角后得到 6°，平分即可得到 3°。

结合两部分问题解决。

时间复杂度 $O(1)$。

DEC14.DIVIDEN

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中最远的点的距离为 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）。

$1 \le T \le 100, 3 \le N \le 400$

算法讨论

按照一些估价函数求出最小生成树，取最优解。

时间复杂度 $O(TN\log N)$。

DEC14.RIN

题目大意

有 N 门课，M 个学期，每门课都需要选择一个学期上课，第 i 门课在第 j 个学期上课可以获得 $A_{i,j}$ 的分数，$A_{i,j} = -1$ 表示无法在第 j 个学期开这门课。还存在 K 组关系，第 i 组关系 u_i,v_i 表示第 u_i 门课必须在第 v_i 门课之前上。询问分数之和的最大值。

数据规模：$1 \le N, M, K \le 100$

算法讨论

使用最小割来解决问题，先将每门课拆成 $M+1$ 个点 $V_{i,1}, V_{i,2}, \cdots, V_{i,m}$，来表示它在哪个学期上课。在 $V_{i,j}$ 与 $V_{i,j+1}$ 之间连边，第 j 个学期上课损失的分数。如果 $A_{i,j} = -1$，要把它的分数变为负无穷。对于每组限制 u,v，在 $V_{u,i}$ 与 $V_{v,i+1}$ 之间连无穷的边。用不管重叠的最大值减去最小割就是答案。

时间复杂度 $O(flow(n, nm))$。

JAN15.RANKA

题目大意

在 9×9 的棋盘上下 N 步棋，使得不存在重复状态，构造一组解。

数据规模：$1 \le N \le 10000$

算法讨论

先构造出 8 个劫，每个劫都可用一个二进制位表示，整个状态就是一个长为 8 的二进制串，每次可以改变一位的状态，用格雷码的方式可以遍历所有的二进制状态。在剩余的位置上每下一子都可以完成一次循环，这样步数就足够了。

时间复杂度 $O(N)$。

JAN15.XRQRS

题目大意

在 9×9 的棋盘上下 N 步围棋，使得不存在重复状态，构造一组解。

算法讨论

先构造出 8 个劫，每个劫都可用一个二进制位表示，整个状态就是一个长为 8 的二进制串，每次可以改变一位的状态，用格雷码的方式可以遍历所有的二进制状态。在剩余的位置上每下一子都可以完成一次循环，这样步数就足够了。

时间复杂度 $O(N)$。

JAN15.SEARD2

题目大意

给出一个整数 A 和 N 个整数 B_i，保证 A 的十进制表示中不含有 0。把 A 中的数字重拍，最小化 $\sum_{i=1}^{N} A\ mod\ B_i$。

数据规模：$1 \le T \le 100, 1 \le N \le 100$

算法讨论

使用爬山算法随机调整数字 A。

时间复杂度 $O(TCN)$，C 表示调整次数。

FEB15.DEVLOCK

题目大意

求有多少 N 位十进制数，满足各位之和小于等于 M，且这个数是 P 的倍数，允许前导 0。

数据规模：$1 \le N \le 10^9, 1 \le P \le 50, 1 \le M \le 15000$。

算法讨论

令 $f_{x,y}$ 表示数模 P 是 x, 数位和是 y 的方案数。考虑到对于 $10^i\ mod\ P$ 相同的数位可以一起考虑。设这样的数位有 t 个，模 P 是 r，用生成函数表示成 $g(x) = (\sum_{i=0}^{9} x^i)^t$，从而 $f_{ir\ mod\ P,i} = [x^i]\, g(x)$。最后需要合并 f 数组，这是一个二维卷积，使用 fft 优化即可。

时间复杂度 $O(MP^3 + MP^2 \log M)$。

FEB15.CUSTPRIM

题目大意

定义三元组 (a, b, c) 和乘法运算，其中 $c = 11$ 或者 $c = 24$。如果令 $(a1, b1, c1)$ 和 $(a2, b2, c2)$ 相乘：

令 $s = (a1a2 + b1b2 + c1c2) + (a1b2 + b1a2) + (c1 + c2), t = \lfloor \frac{s}{2} \rfloor + 16(c1 + c2) - c1c2$

$A = (t - 2(a1b2 + b1a2) - (a1c2 - c1a2) + 33(a1 + a2) + (b1b2 - a1a2)$

$B = (t - 5(a1b2 + b1a2) - (c1b2 + b1c2) + 33(b1 + b2) + (2b1b2 + 4a1a2)$

如果 s 是偶数，那么结果就是 $(A - 540, B - 540, 24)$，否则结果就是 $(A - 533, B - 533, 11)$。

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。

给一个三元组，判断是否为质数。

数据规模：$1 \le T \le 10^4, -10^7 \le a, b \le 10^7$。

算法讨论

令 ω 是满足方程 $\omega^2 - \omega + 3 = 0$ 的解，即 $\omega = \frac{1+\sqrt{-11}}{2}$，问题转化成判断域 $Z[\omega]$ 下的数 $a + b\omega$ 是否是质数。定义 $(a + b\omega)' = (a + b - b\omega)$，那么令 $Nx = xx'$，有如下结论：

1. 如果 x 不是整数，那么 x 是质数当且仅当 Nx 是质数。

2. 如果 x 是整数，那么 x 当且仅当 x 是质数且 $\mid x \mid = 2$ 或 $\mid x \mid \neq 11$ 且 -11 在模 x 下没有二次剩余。

使用 Miller Rabin 判断。

FEB15.CUSTPRIM continued on next page...

时间复杂度 $O(T\log\mid a\mid)$。

MARCH15.TREECNT2

题目大意

给出一棵 N 个节点的树，支持两种操作：

1. 修改一条边的权值。

2. 询问有多少条路径满足路径上边权的最小公约数是 1。

数据规模：$1\leq N\leq 10^5$，$1\leq Q\leq 100$，边权 V 不超过 10^6。

算法讨论

询问中需要满足最小公约数恰好为 1，如果计算出所有最小公约数为 i 的倍数的路径数量 cnt_i，用莫比乌斯函数累加就可以了。

考虑如何计算 cnt_i，先不管修改，把是 i 倍数的边称为有效边，这些有效边会组成森林。可以发现，在同一棵树中的任意两点间的路径都是合法路径。那么只需要维护连通块大小，如果仅有加边，可以用并查集维护。但存在修改操作时，边权改变导致有效边改变，可以用 cdq 分治来使得只存在加边和撤销操作，用栈来记录并查集的修改内容即可。

时间复杂度 $O(tot(n+Q\log Q)\log n+V)$，tot 为边权的因数个数。

MARCH15.RNG

题目大意

给定一个数列的递推式：$A_i=\sum_{j=1}^{k}C_j\times A_{i-j}\ mod\ 104857601$。求 A_n。

数据规模：$1\leq nleq10^18,k\leq 3\times 10^4$。

算法讨论

首先这个递推式是线性的，但直接矩乘复杂度太高。求出这个矩阵的特征多项式 $f(x)=x^k-\sum_{i=1}^{k}C_kx^{k-1}$，令 $g(x)=x^n\ mod\ f(x)$，答案可以表示成 $\sum_{i=1}^{k}[x^{i-1}]\,g(x)\times C_i$。使用多项式取模算法来计算。

时间复杂度 $O(k\log k\log n)$。

APRIL15.BWGAME

题目大意

初始一个 $N*N$ 的矩阵，第 i 行的第 L_i 列到第 R_i 列是黑色的，其余为白色。对于一个排列 P，合法当且仅当所有 (i,P_i) 都为黑色。求合法排列逆序对个数奇数多还是偶数多。

数据规模：$1\leq N\leq 10^5$。

算法讨论

考虑在矩阵的最后一行选一个数，那么相当于删去了一行一列，剩余部分又组成了一个矩阵。如果选择的数在第 i 列，那么就会增加 $n-i$ 个逆序对。再考虑交换相邻两列，相当于每一个排列的这两个位置都交换了，逆序对个数都相差了 1，也就是奇变偶，偶变奇。由此可以发现这就是一个行列式。那么可以用高斯消元来解决计算这个行列式的值。直接消元是 n^2 的，可以用优先队列来优化。

时间复杂度 $O(n\log n)$。

APRIL15.LPARTY

题目大意

APRIL15.LPARTY continued on next page...

有 M 个长为 N 的字符串，第 i 位是字典序为 i 的大小写字母。可以把一些串并成它们的公共子序列，但需要取遍所有情况。求并出来的最短长度。

数据规模：$1 \le N \le 5$，$1 \le M \le 1000$。

算法讨论

不同的串最多有 32 个。可以枚举子序列，用 Bit 来表示它包含的串状态，Len 来表示它的长度。如果存在 i, j，$Bit_i \ and \ Bit_j = Bit_j$，$Len_i < Len_j$。那么 j 是个无效的子序列。删去所有无效子序列后使用搜索加剪枝就可以找到答案。

时间复杂度 $O(2^{state})$。

MAY15.CBAL

题目大意

一个字符串被称作平衡的，当且仅当每个字母都出现了偶数次。回答 Q 组询问，每组询问 L,R,T，表示求 $[L, R]$ 这个字符串的所有平衡子串的长度的 T 次方和。

数据规模：$1 \le N, Q \le 10^5$，$T \in \{0, 1, 2\}$。

算法讨论

用二进制数 pre_i 来记录 $[1, i]$ 这个字符串中每个字母的奇偶性，那么两个相同的二进制数之间就会产生一个平衡字符串。按照下标分块后，统计前 i 个块中状态是 j 的下标数量、和以及平方和；某个状态在前 i 个块中出现的次数。对于一个询问，先得到中间连续块中的答案，再逐个加入两边的状态计算贡献。

时间复杂度 $O(N\sqrt{N})$。

MAY15.GRAPHCNT

题目大意

给出 N 个点 M 条边的有向图，统计 (x, y) 的数量满足存在一条 1 到 x 的路径和一条 1 到 y 的路径，使得这两条路径在除 1 点以外没有交。

数据规模：$1 \le N \le 10^5$ $1 \le M \le 5 \times 10^5$。

算法讨论

先用 dominator tree 计算出每个点的必经点，并把这个点当做父亲建出树。在树上自底向上统计一遍就能得到答案。

时间复杂度 $O(N)$。

JUNE15.CHEFBOOK

题目大意

给定 M 个整数对 (x, y)，$(1 \le x, y \le N)$ 和对应的 $L_{x,y}$、$S_{x,y}$ 和 $T_{x,y}$。需要求出 N 个非负整数 P_X，和 N 个非负整数 Q_x，使得 $S_{x,y} \le L_{x,y} + P_x - Q_y \le T_{x,y}$，并且 $L_{x,y} + P_x - Q_y$ 最大。

数据规模：$1 \le N \le 100, 0 \le M \le N^2, 0 \le P_X, Q_x \le 10^6$。

算法讨论

每条限制都可以表示成两个变量的差小于等于某个常数，考虑用网络流来处理。把变量当做点，两个相关变量 x，y，满足 $x - y \le C$，就在 x 点和 y 点间连流量无穷，费用为 C 的边。如果变量 x 在方程中出现了 t 次，且是 P 类变量，就在源和 x 之间连流量为 t，费用为 0 的边；如果是 Q 类变量就和汇连。这样连边后流量平衡的关系和费用的和满足了问题的需求。

最后还需要求出一组解，先按残余网络上流量的情况连出有效的边，再在上面跑差分约束系统就可以得到解了。

JUNE15.CHEFBOOK continued on next page...

时间复杂度 $O(flow(N,M)+NM^2)$。

JULY15.EASYEX

题目大意

假设你有一个 K 面骰子，上面写着 1 到 K。还有两个整数 L 和 F($0<L\le K$)。你掷了 N 次骰子，记掷出数字 i 的次数为 a_i。

请你求出 $a_1^F\times a_2^F\times\cdots\times a_L^F$ 的期望值。

数据规模：$0\le N,K\le 10^9$，$0<F\le 1000$，$0<L\times F\le 50000$。

算法讨论

把 a_i 表示成 $v_{i,1}+v_{i,2}+\ldots+v_{i,n}$，其中 $v_{i,n}$ 表示第 n 次掷出的数字是否是 i。把式子展开就变成了连加，根据期望的线性性，可以分别算每一项的贡献。由于每一次只能掷出一个数字，如果一项中存在 $v_{x,z}$ 和 $v_{y,z}$，且 $x\neq y$，这一项的贡献就是 0，否则就是 $\frac{1}{K^{num}}$, 其中 num 表示变量数。

先算出 a_u^F 展开后的情况。用 $pre_{i,j}$ 表示 a_u^i 展开后 j 个变量的系数和。可以用

$$pre_{i,j}=pre_{i-1,j-1}+pre_{i-1,j}*j$$

来递推。这个也就是第二类 $stirling$ 数。

最后 dp 合并：

$$dp_{i,j}=dp_{i-1,k}\times\binom{j}{k}\times pre_{F,j-k}$$

考虑到这个转移是个卷积，可以用 FFT+ 快速幂解决。

时间复杂度 $O(F^2+LF\log(LF))$

JULY15.HAMILG

题目大意

给定一个无向图，两个人玩游戏。第一个人选择一个出发点，第二个人开始两个玩家轮流操作，每次操作可以沿着边移动到没有到达过的点。无法移动者输。求多少个出发点第一个人必胜。

数据规模：$1\le N\le 2000, 1\le M\le 3\times 10^5$。

算法讨论

在问题所用图是二分图时，是个经典问题，只需要二分图匹配后遍历一遍。当问题是一般图时，考虑用带花树来代替二分图，匹配完后找出孤立点再找增广路。

时间复杂度 $O(NM)$

AUG15.CLOWAY

题目大意

给出 T 个 N 点无向无权图。在第 i 次询问中，给出三个参数 L_i,R_i,K_i，同时在标号 $[L_i,R_i]$ 的无向图中进行游戏。

对于 L_i 到 R_i 的无向图，先随机一个起点，在之后的每一步中选取一个随机非空集合，在所有被选的无向图中随机走一步。当所有点回到起点时游戏结束。

对于每一次询问，回答有多少种方案使得它在 K_i 步内结束。

数据规模：$1\le N,T\le 50, 1\le K_i\le 10000, 1\le Q\le 10^5$。

算法讨论

首先求出每个图的邻接矩阵 G_i，如果只有一张图，那么只需要将这张图的矩阵 K 次方之后求出对角线上数的和，也就是环的数量就可以了。用 $tr[G]$ 来表示矩阵 G 对角线上数字和，这又叫矩阵的迹。

矩阵的迹等于矩阵的谱中元素的和，用 $sp(G)$ 来表示矩阵 G 的谱。多个图的情况下问题其实是求这些图的强连通图对应的矩阵的迹，而两个矩阵组成的强连通图的谱可以表示成：

$$sp(G \times H) = \{(\alpha + 1)(\beta + 1) - 1 \mid \alpha \in sp(G), \beta \in sp(H)\}$$

由此可以得到

$$tr[(G \times H)^k] = \sum_{i=0}^{k} \binom{k}{i} (-1)^{k-i} tr((G + I)^k) tr((H + I)^k)$$

$$tr[(G_l \dots G_r)^k] = \sum_{i=0}^{k} \binom{k}{i} (-1)^{k-i} tr[(G_l + I)^k] \dots tr[(G_r + I)^k]$$

这能表示成卷积的形式，用 fft 加速运算。但问题的模数是 $10^9 + 7$，不能直接使用 ntt，可以选择一些较大的模数 ntt 后用中国剩余定理合并。

时间复杂度 $O(TN^4 + TNK + T^2 K \log K)$。

AUG15.DISTNUM

题目大意

给定一个长度为 N 的数列和 M 个操作。

1. 定义 S 为区间 $[L, R]$ 中出现过的数字的集合。求 $\sum_{1 \le i < j < k \le |S|} S_i S_j S_k$。
2. 插入一个数。
3. 删除一个数。
4. 修改一个数。
5. 询问一个区间内出现过的数字种类数。

数据规模：$1 \le N, M \le 10^5$。

算法讨论

在同一个区间内出现多个相同的数应该算作一个，可以通过同时记录当前位置和下一个相同数字出现位置，转化为二维问题解决。插入删除数字可以先离线加入后给每个数一个固定的位置。询问只需要求出所有数的零至三次方和，这只需要树套树就可以解决了。

时间复杂度 $O(M \log^2 N)$。