多项式与线性代数

yanQval

2018 年 1 月 24 日

前言

今天的讲课的题目其实是~~多项式~~与线性代数。

前言

今天的讲课的题目其实是~~多项式~~与线性代数。

在开始讲课之前相信大家都会：

前言

今天的讲课的题目其实是~~多项式~~与线性代数。

在开始讲课之前相信大家都会：

- FFT 和 NTT

前言

今天的讲课的题目其实是~~多项式~~与线性代数。
在开始讲课之前相信大家都会：

- FFT 和 NTT
- 多项式求逆

前言

今天的讲课的题目其实是~~多项式~~与线性代数。

在开始讲课之前相信大家都会：

- FFT 和 NTT
- 多项式求逆
- 多项式除法取模，多项式多点求值

前言

今天的讲课的题目其实是~~多项式~~与线性代数。

在开始讲课之前相信大家都会：

- FFT 和 NTT
- 多项式求逆
- 多项式除法取模，多项式多点求值
- ...

矩阵的一些基础定义

矩阵的一些基础定义

A_{nm} 表示一个 n 行 m 列的矩阵。

特别的，一个 1 行 n 列的矩阵，可以被称为行向量，列向量同理。

一个 n 行 n 列的矩阵，可以被称为 n 阶方阵 A_n。

A^T 表示矩阵 A 的转置，即 $a_{ij}^T = a_{ji}$，相当于把矩阵沿主对角线翻转。

对角矩阵表示除了主对角线上的元素全部为 0 的矩阵。

主对角线以下全部为 0 的方阵是上三角矩阵。

单位矩阵是主对角线上面元素全部为 1 的对角矩阵，一般用 I 表示。

矩阵乘法

矩阵乘法大家都会，我们来看两个题目。

Arithmetic of Bomb 2

Arithmetic of Bomb 2

给定一个只含有数字, 加号减号乘号的表达式，请计算表达式的值, 答案对 10^9+7 取模。

表达式可能会有一段重复若干次，用 (x)#(y) 表示，含义是表达式 x 重复 y 次，重复不存嵌套关系。

比如 (1-2+3)#(3) 展开之后应该是 1-2+31-2+31-2+3。

令 N= 表达式长，M= # 后的括号内的数字的最大值。

$N \leq 2*10^6; M \leq 10^{18}$。

Arithmetic of Bomb 2

Arithmetic of Bomb 2

我们考虑顺序计算这个表达式。假设当前表达式为 $a + b*c$，我们用三元组 (a,b,c) 表示。

Arithmetic of Bomb 2

我们考虑顺序计算这个表达式。假设当前表达式为 $a + b*c$，我们用三元组 (a,b,c) 表示。

假如加入了一个数字 d，表达式变为 $a + b*\overline{cd} = a + b*(c*10 + d)$，即 (a,b,c*10+d)

Arithmetic of Bomb 2

我们考虑顺序计算这个表达式。假设当前表达式为 $a + b*c$，我们用三元组 (a,b,c) 表示。

假如加入了一个数字 d，表达式变为 $a + b*\overline{cd} = a + b*(c*10 + d)$，即 (a,b,c*10+d)

假如加入了一个乘号，表达式变为 $a + b*c*0$，即 (a,b*c,0)

Arithmetic of Bomb 2

我们考虑顺序计算这个表达式。假设当前表达式为 $a + b*c$，我们用三元组 (a,b,c) 表示。

假如加入了一个数字 d，表达式变为 $a + b*\overline{cd} = a + b*(c*10 + d)$，即 (a,b,c*10+d)

假如加入了一个乘号，表达式变为 $a + b*c*0$，即 (a,b*c,0)

假如加入了一个加号，表达式变为 $a + b*c + 1*0$，即 (a+b*c,1,0)

Arithmetic of Bomb 2

我们考虑顺序计算这个表达式。假设当前表达式为 $a + b{*}c$，我们用三元组 (a,b,c) 表示。

假如加入了一个数字 d，表达式变为 $a + b{*}\overline{cd} = a + b{*}(c{*}10 + d)$，即 (a,b,c*10+d)

假如加入了一个乘号，表达式变为 $a + b{*}c{*}0$，即 (a,b*c,0)

假如加入了一个加号，表达式变为 $a + b{*}c + 1{*}0$，即 (a+b*c,1,0)

假如加入了一个减号，表达式变为 $a + b{*}c{-}1{*}0$，即 (a+b*c,-1,0)

Arithmetic of Bomb 2

我们考虑顺序计算这个表达式。假设当前表达式为 $a+b*c$，我们用三元组 (a,b,c) 表示。

假如加入了一个数字 d，表达式变为 $a+b*\overline{cd}=a+b*(c*10+d)$，即 (a,b,c*10+d)

假如加入了一个乘号，表达式变为 $a+b*c*0$，即 (a,b*c,0)

假如加入了一个加号，表达式变为 $a+b*c+1*0$，即 (a+b*c,1,0)

假如加入了一个减号，表达式变为 $a+b*c-1*0$，即 (a+b*c,-1,0)

注意到以上的转移，全部是关于 a,b,b*c,1 的线性变换，所以我们可以通过维护一个长度的为 4 的行向量来解决这个问题。每次遇到一个重复式子，就把前面括号内的矩阵整体求出来，然后快速幂求得展开之后的矩阵。

小 Y 和恐怖的奴隶主

小 Y 正在打一个 BOSS。

BOSS 只带了一个具有 m 点生命值的恐怖的奴隶主。

每当一个恐怖的奴隶主收到伤害而没有死亡，而 BOSS 的随从数量小于上限 k，就会召唤一个新的具有 m 点生命值的恐怖的奴隶主。

小 Y 会攻击 n 次，每次会从 BOSS 所有的随从和 BOSS 本身中随机选取一个扣减生命值 1，求 n 次攻击之后，期望下对 BOSS 造成了多少伤害。每组数据给定了 m 和 k，然后询问 T 次 n。

$T \leq 500, n \leq 10^{18}, m \leq 3, k \leq 8$

小 Y 和恐怖的奴隶主

容易发现场面只与现在血量为 0 到 m 的奴隶主各有多少个有关。显然这个数字最大为 $\binom{8+3}{3} = 165$

小 Y 和恐怖的奴隶主

容易发现场面只与现在血量为 0 到 m 的奴隶主各有多少个有关。显然这个数字最大为 $\binom{8+3}{3} = 165$

令 f[i][j] 表示 i 步之后，现在场面是 j 的概率，j 是一个 1 到 165 的数字，令 gi 表示 i 步之后的期望伤害。

小 Y 和恐怖的奴隶主

容易发现场面只与现在血量为 0 到 m 的奴隶主各有多少个有关。显然这个数字最大为 $\binom{8+3}{3} = 165$

令 f[i][j] 表示 i 步之后，现在场面是 j 的概率，j 是一个 1 到 165 的数字，令 gi 表示 i 步之后的期望伤害。

令 N = 场面数 +1，显然可以得到一个 $TN^3 \log n$ 的做法。

小 Y 和恐怖的奴隶主

容易发现场面只与现在血量为 0 到 m 的奴隶主各有多少个有关。显然这个数字最大为 $\binom{8+3}{3} = 165$

令 f[i][j] 表示 i 步之后，现在场面是 j 的概率，j 是一个 1 到 165 的数字，令 gi 表示 i 步之后的期望伤害。

令 N = 场面数 +1，显然可以得到一个 $TN^3 \log n$ 的做法。

令转移矩阵为 M，注意到我们每次的瓶颈在于求 M^n，而我们要求的是一个行向量 V (初始概率) 乘上这个矩阵的结果。

小 Y 和恐怖的奴隶主

容易发现场面只与现在血量为 0 到 m 的奴隶主各有多少个有关。显然这个数字最大为 $\binom{8+3}{3} = 165$

令 f[i][j] 表示 i 步之后，现在场面是 j 的概率，j 是一个 1 到 165 的数字，令 gi 表示 i 步之后的期望伤害。

令 N = 场面数 +1，显然可以得到一个 $TN^3 \log n$ 的做法。

令转移矩阵为 M，注意到我们每次的瓶颈在于求 M^n，而我们要求的是一个行向量 V (初始概率) 乘上这个矩阵的结果。

矩阵具有结合律，我们没有必要一定要求得 Mn。我们可以通过预处理得到 M^0, M^1, M^2, M^4, 之后每次的询问，我们只要选取 $\log n$ 的矩阵与 V 相乘即可，复杂度 $O(N^3 \log N + TN^2 \log n)$

逆矩阵

矩阵 A 的逆矩阵 A^{-1}, 是满足 $AA^{-1} = A^{-1}A = I$ 的矩阵

逆矩阵

矩阵 A 的逆矩阵 A^{-1}, 是满足 $AA^{-1} = A^{-1}A = I$ 的矩阵

并不是每个矩阵都是存在逆矩阵的，存在的条件后面再讨论

逆矩阵

矩阵 A 的逆矩阵 A^{-1}, 是满足 $AA^{-1} = A^{-1}A = I$ 的矩阵

并不是每个矩阵都是存在逆矩阵的，存在的条件后面再讨论

求解逆矩阵的方法：

我们将原矩阵的右侧放上一个单位矩阵，然后对整体进行消元，当左边被消成单位矩阵的时候，右侧就得到了逆矩阵。如果中途失败，说明原矩阵不可逆。

逆矩阵

矩阵 A 的逆矩阵 A^{-1}, 是满足 $AA^{-1} = A^{-1}A = I$ 的矩阵

并不是每个矩阵都是存在逆矩阵的，存在的条件后面再讨论

求解逆矩阵的方法：

我们将原矩阵的右侧放上一个单位矩阵，然后对整体进行消元，当左边被消成单位矩阵的时候，右侧就得到了逆矩阵。如果中途失败，说明原矩阵不可逆。

证明：考虑消元使用的矩阵初等行变换，都是可以表示成左乘一个矩阵的，在把左侧消成单位矩阵的过程中得到的初等变换的矩阵的乘积，就是逆矩阵。通过构造右边放一个矩阵，来收集这些操作即可。

行列式

定义:

一个方阵的行列式表示为 $|A|$

$$|A| = \sum_{p} (-1)^{\sigma(p)} \prod_{i=1}^{n} a_{i,p_i}$$

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

交换矩阵的两行/两列，行列式值取反

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

交换矩阵的两行/两列，行列式值取反

证明：对于一个排列，交换任意两个元素，排列的逆序对奇偶性一定改变。

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

交换矩阵的两行/两列，行列式值取反

证明：对于一个排列，交换任意两个元素，排列的逆序对奇偶性一定改变。

将矩阵的一行/一列乘上一个固定的常数 k，行列式值也乘上 k。

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

交换矩阵的两行/两列，行列式值取反

证明：对于一个排列，交换任意两个元素，排列的逆序对奇偶性一定改变。

将矩阵的一行/一列乘上一个固定的常数 k，行列式值也乘上 k。

证明非常显然。

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

交换矩阵的两行/两列，行列式值取反

证明：对于一个排列，交换任意两个元素，排列的逆序对奇偶性一定改变。

将矩阵的一行/一列乘上一个固定的常数 k，行列式值也乘上 k。

证明非常显然。

将矩阵的一行加到另外一行上去，行列式值不变，列同理。

行列式性质

基础性质：

一个对角矩阵/上三角矩阵的行列式值是所有对角线上元素的乘积。

证明非常显然。

交换矩阵的两行/两列，行列式值取反

证明：对于一个排列，交换任意两个元素，排列的逆序对奇偶性一定改变。

将矩阵的一行/一列乘上一个固定的常数 k，行列式值也乘上 k。

证明非常显然。

将矩阵的一行加到另外一行上去，行列式值不变，列同理。

证明我们先来考虑几个更简单的结论，以下证明只考虑行变换，列变换同理。

行列式性质

存在两行完全一样的矩阵，行列式值为 0。

行列式性质

存在两行完全一样的矩阵，行列式值为 0。

证明：假设第 x 行和第 y 行完全一样，那么交换排列中的 p_x 和 p_y，$\prod a_{i,p_i}$ 不变，而前面的系数一定取反，所以行列式的每一项都存在一项和它绝对值一样，符号相反。

行列式性质

存在两行完全一样的矩阵，行列式值为 0。

证明：假设第 x 行和第 y 行完全一样，那么交换排列中的 p_x 和 p_y，$\prod a_{i,p_i}$ 不变，而前面的系数一定取反，所以行列式的每一项都存在一项和它绝对值一样，符号相反。

对于一个矩阵 A 确定的一行 x，令 a_i 表示第 x 行第 i 列的元素。假设对于每一个 i，都满足 $b_i + c_i = a_i$。那么构造两个矩阵 B 和 C，使得矩阵 B 第 x 行第 i 列是 b_i，C 同理。那么 $|A| = |B| + |C|$

行列式性质

存在两行完全一样的矩阵，行列式值为 0。

证明：假设第 x 行和第 y 行完全一样，那么交换排列中的 p_x 和 p_y，$\prod a_{i,p_i}$ 不变，而前面的系数一定取反，所以行列式的每一项都存在一项和它绝对值一样，符号相反。

对于一个矩阵 A 确定的一行 x，令 a_i 表示第 x 行第 i 列的元素。假设对于每一个 i，都满足 $b_i + c_i = a_i$。那么构造两个矩阵 B 和 C，使得矩阵 B 第 x 行第 i 列是 b_i，C 同理。那么 $|A| = |B| + |C|$

证明：直接使用定义式即可。

行列式性质

重新考虑之前的那个变换，把一行加到另外一行上去，可以把得到的矩阵这一行拆成原本的和加上去的部分，由于加上去的部分行列式值一定为 0，所以行列式值不变。

以上三个变换就是行列式的初等变换。

行列式性质

重新考虑之前的那个变换，把一行加到另外一行上去，可以把得到的矩阵这一行拆成原本的和加上去的部分，由于加上去的部分行列式值一定为 0，所以行列式值不变。

以上三个变换就是行列式的初等变换。

根据这三个初等变换，我们可以通过高斯消元在 $O(n^3)$ 的复杂度内求得行列式的值。

行列式性质

行列式性质

这里也可以证明，矩阵可逆的充要条件是行列式不为 0。

因为行列式为 0，说明消元过程中出现了 $a_{ii}=0$，也就是上文提到的中途失败。

运河计划

给定一个 n 行 m 列的网格图，网格图上有 q 个坏点不能经过，现在网格图的最上方有 p 个人，他们的位置分别在 $(0, a_1), (0, a_2) \ldots (0, a_p)$, 他们的目标是最下方的 p 个点，位置分别是 $(n, b_1), (n, b_2) \ldots (n, b_p)$。

保证 $a_1 \le a_2 \le \ldots \le a_p, b_1 \le b_2 \le \ldots \le b_p$, 每个人只能向下和向右走, 每个点只能被一个人经过，求合法的方案数，答案对 998244353 取模

$n, m \le 100000, p, q \le 200$

运河计划

考虑经典容斥，枚举哪些人的路径相交了。

运河计划

考虑经典容斥，枚举哪些人的路径相交了。

相交的路径可以转化为，在相交点将两个人之后的路径取反，最终是第一个人走到第二个人的终点去了，第二个人走到第一个人的终点前去了。

运河计划

考虑经典容斥，枚举哪些人的路径相交了。

相交的路径可以转化为，在相交点将两个人之后的路径取反，最终是第一个人走到第二个人的终点去了，第二个人走到第一个人的终点前去了。

直接这样容斥复杂度是 $O(n!)$ 的。

运河计划

考虑经典容斥，枚举哪些人的路径相交了。

相交的路径可以转化为，在相交点将两个人之后的路径取反，最终是第一个人走到第二个人的终点去了，第二个人走到第一个人的终点前去了。

直接这样容斥复杂度是 O(n!) 的。

进一步观察可以发现，容斥的系数就是逆序对的奇偶性，这和行列式的定义完全一样。

运河计划

考虑经典容斥，枚举哪些人的路径相交了。

相交的路径可以转化为，在相交点将两个人之后的路径取反，最终是第一个人走到第二个人的终点去了，第二个人走到第一个人的终点前去了。

直接这样容斥复杂度是 $O(n!)$ 的。

进一步观察可以发现，容斥的系数就是逆序对的奇偶性，这和行列式的定义完全一样。

我们可以求出每个起点到每个终点的方案数，组成一个矩阵，求出来行列式就是答案。

运河计划

考虑经典容斥，枚举哪些人的路径相交了。

相交的路径可以转化为，在相交点将两个人之后的路径取反，最终是第一个人走到第二个人的终点去了，第二个人走到第一个人的终点前去了。

直接这样容斥复杂度是 $O(n!)$ 的。

进一步观察可以发现，容斥的系数就是逆序对的奇偶性，这和行列式的定义完全一样。

我们可以求出每个起点到每个终点的方案数，组成一个矩阵，求出来行列式就是答案。

求出每个点到每个点的方案数，使用经典容斥的方法。最终复杂度 $O(pq^2 + p^3)$。

模范学长

给定一个 N 阶方阵 A, 每个元素都是一个整系数多元多项式，判断这个方阵的行列式，在合并同类项之后，是否每项之前的系数都是偶数

数据组数 $\leq 20, N \leq 50$，变量个数不超过 26, 每个元素的表达式长度不超过 100

模范学长

如果多项式的系数全部为偶数。那么显然这个多项式带入什么值都是偶数。一种直观的想法是给每个变量随机一个取值，带入检验答案是否是偶数

模范学长

如果多项式的系数全部为偶数。那么显然这个多项式带入什么值都是偶数。一种直观的想法是给每个变量随机一个取值，带入检验答案是否是偶数

不过这种想法显然是错的，考虑 $x+x^2$ 无论带入什么都是 0。

模范学长

如果多项式的系数全部为偶数。那么显然这个多项式带入什么值都是偶数。一种直观的想法是给每个变量随机一个取值，带入检验答案是否是偶数

不过这种想法显然是错的，考虑 $x + x^2$ 无论带入什么都是 0。

Schwartz-Zippel 定理说明了有限域 F 上的 d 次非零多元多项式 $f(x1, x2, \ldots x_n)$，随机代入 x_i 的值，多项式值为 0 的概率是不超过 $\frac{d}{|F|}$ 的。

模范学长

如果多项式的系数全部为偶数。那么显然这个多项式带入什么值都是偶数。一种直观的想法是给每个变量随机一个取值，带入检验答案是否是偶数

不过这种想法显然是错的，考虑 $x+x^2$ 无论带入什么都是 0。

Schwartz-Zippel 定理说明了有限域 F 上的 d 次非零多元多项式 $f(x1, x2, \ldots x_n)$，随机代入 x_i 的值，多项式值为 0 的概率是不超过 $\frac{d}{|F|}$ 的。

需要的是寻找一个合适的有限域 F，满足：

- F 的特征为 2
- |F| 足够大

模范学长

我们考虑构造一个特征为 2 的有限域

模范学长

我们考虑构造一个特征为 2 的有限域

模 p 意义下剩余系的特征为 p, 所以只有模 2 可行

模范学长

我们考虑构造一个特征为 2 的有限域

模 p 意义下剩余系的特征为 p, 所以只有模 2 可行

考虑使用多项式环来构造，构造系数模 2 意义下 t 次不可约多项式 P(x)，那么模 P(x) 意义下的剩余系就是一个特征为 2 的域。并且这个域的大小至少为 2^{t-1}。

模范学长

我们考虑构造一个特征为 2 的有限域

模 p 意义下剩余系的特征为 p, 所以只有模 2 可行

考虑使用多项式环来构造，构造系数模 2 意义下 t 次不可约多项式 P(x)，那么模 P(x) 意义下的剩余系就是一个特征为 2 的域。并且这个域的大小至少为 2^{t-1}。

于是我们可以多次给每个变量随机一个多项式，然后求行列式即可，随机多次正确率非常高。

模范学长

我们考虑构造一个特征为 2 的有限域

模 p 意义下剩余系的特征为 p, 所以只有模 2 可行

考虑使用多项式环来构造，构造系数模 2 意义下 t 次不可约多项式 P(x)，那么模 P(x) 意义下的剩余系就是一个特征为 2 的域。并且这个域的大小至少为 2^{t-1}。

于是我们可以多次给每个变量随机一个多项式，然后求行列式即可，随机多次正确率非常高。

高斯消元求行列式的时候涉及到的除法操作实现不是很方便，有两种方法解决，一是辗转相除，二是我们只要判断行列式是否为 0，所以可以直接不用除。

一些定义

余子式: 将方阵的第 i 行和第 j 列同时划去，剩余的一个 n - 1 阶的矩阵的行列式值称为元素 a_{ij} 的余子式，通常记为 M_{ij}

一些定义

余子式: 将方阵的第 i 行和第 j 列同时划去，剩余的一个 n - 1 阶的矩阵的行列式值称为元素 a_{ij} 的余子式，通常记为 M_{ij}

代数余子式：元素 a_{ij} 的代数余子式是 $C_{ij} = (-1)^{i+j}M_{ij}$

拉普拉斯展开

对于一个方阵 A，A 的行列式相等于某一行所有元素的值乘上他的代数余子式的和。

即：$|A| = \sum_{i=1}^{n} a_{xi} * C_{xi}$，x 是一个确定的行坐标。对于某一列的展开同理。

证明考虑直接展开定义式

拉普拉斯展开提供了降低行列式阶数的方法，是化简行列式的常见方法。

伴随矩阵

矩阵 A 的代数余子矩阵是由每个元素的代数余子式构成的矩阵。

伴随矩阵

矩阵 A 的代数余子矩阵是由每个元素的代数余子式构成的矩阵。
矩阵 A 的伴随矩阵 A* 是 A 的代数余子矩阵的转置。即 $A* = C^T$。

伴随矩阵

矩阵 A 的代数余子矩阵是由每个元素的代数余子式构成的矩阵。

矩阵 A 的伴随矩阵 A* 是 A 的代数余子矩阵的转置。即 $A* = C^T$。

对于可逆矩阵，满足：

$$A* = |A|A^{-1}$$

伴随矩阵

矩阵 A 的代数余子矩阵是由每个元素的代数余子式构成的矩阵。

矩阵 A 的伴随矩阵 A* 是 A 的代数余子矩阵的转置。即 $A* = C^T$。

对于可逆矩阵，满足：

$$A* = |A|A^{-1}$$

证明：考虑证明 $AA* = |A|I$

对于满足 $i = j$ 的元素，显然是 $\sum_{k=1}^{n} a_{ik}C_{ik}$，根据拉普拉斯展开，这个值恰好是 $|A|$。

对于满足 $i \neq j$ 的元素，显然是 $\sum_{k=1}^{n} a_{ik}C_{jk}$，这个等价于把第 j 行换为第 i 行的矩阵，显然行列式为 0。

高等代数

给定一个 $n+1$ 行 n 列的矩阵，满足矩阵的每一列从上到下都是一个从 1 开始的等比数列

给定每一列的公比 x_i, 问删去每一行之后矩阵的行列式值

$n \le 100000, x_i \le 200000$

高等代数

矩阵的每一列都是一个等比数列的行列式比较特殊，其实是 Vandermonde 行列式。

高等代数

矩阵的每一列都是一个等比数列的行列式比较特殊，其实是 Vandermonde 行列式。

我们可以尝试手动归纳一下 Vandermonde 行列式的值，可以轻易发现是 $\prod_{i<j} x_j - x_i$。

高等代数

矩阵的每一列都是一个等比数列的行列式比较特殊，其实是 Vandermonde 行列式。

我们可以尝试手动归纳一下 Vandermonde 行列式的值，可以轻易发现是 $\prod_{i<j} x_j - x_i$。

直接做不太方便做，我们可以考虑添加上 $n+1$ 列，变为一个完整的 Vandermonde 行列式

高等代数

矩阵的每一列都是一个等比数列的行列式比较特殊，其实是 Vandermonde 行列式。

我们可以尝试手动归纳一下 Vandermonde 行列式的值，可以轻易发现是 $\prod_{i<j} x_j - x_i$。

直接做不太方便做，我们可以考虑添加上 $n+1$ 列，变为一个完整的 Vandermonde 行列式

这时候答案变成了求所有的 $M_{x,n+1}$

高等代数

矩阵的每一列都是一个等比数列的行列式比较特殊，其实是 Vandermonde 行列式。

我们可以尝试手动归纳一下 Vandermonde 行列式的值，可以轻易发现是 $\prod_{i<j} x_j - x_i$。

直接做不太方便做，我们可以考虑添加上 $n+1$ 列，变为一个完整的 Vandermonde 行列式

这时候答案变成了求所有的 $M_{x,n+1}$

假设我们添加上的第 $n+1$ 列是一个变量元 x, 那么根据归纳出来的公式考虑现在的行列式，应为 $\prod\limits_{i<j\leq n}(x_j - x_i)\prod_{i=1}^{n}(x - x_i)$, 展开之后应当为一个 n 次多项式

高等代数

重新从拉普拉斯展开的角度考虑这个行列式值。

高等代数

重新从拉普拉斯展开的角度考虑这个行列式值。

显然展开最后一列可以得到，行列式值为 $\sum_{i=1}^{n+1} C_{i,n+1}x^{i-1}$，还是一个 n 次多项式。

高等代数

重新从拉普拉斯展开的角度考虑这个行列式值。

显然展开最后一列可以得到，行列式值为 $\sum_{i=1}^{n+1} C_{i,n+1}x^{i-1}$，还是一个 n 次多项式。

注意到第二个多项式的系数直接对应答案，所以我们直接展开第一个多项式即可得到答案。

高等代数

重新从拉普拉斯展开的角度考虑这个行列式值。

显然展开最后一列可以得到，行列式值为 $\sum_{i=1}^{n+1} C_{i,n+1}x^{i-1}$，还是一个 n 次多项式。

注意到第二个多项式的系数直接对应答案，所以我们直接展开第一个多项式即可得到答案。

展开使用分治 fft 解决。还有一个问题就是如何计算 $\prod x_j - x_i$。这道题的值域范围不大，可以直接一次 fft 解决。如果值域范围过大，可以考虑使用分治之后进行多项式多点求值

permutation

你要生成一个长度为 n 的排列，有 m 个可行数对，每个数对形如" bi 这个数字可以放在第 ai 个位置上"。

已知现在能够生成出来的排列的数量是奇数个。你想知道删去每一个数对之后剩余的数对能够生成的排列的数的奇偶性。

$n \leq 2000$

permutation

你要生成一个长度为 n 的排列，有 m 个可行数对，每个数对形如" bi 这个数字可以放在第 ai 个位置上"。

已知现在能够生成出来的排列的数量是奇数个。你想知道删去每一个数对之后剩余的数对能够生成的排列的数的奇偶性。

$n \leq 2000$

permutation

如果我们生成一个矩阵 $a_{i,j} = [$数对 (i, j) 存在$]$, 那么答案显然是 $\sum_p \prod_{i=1}^n a_{ip_i}$

permutation

如果我们生成一个矩阵 $a_{i,j} = $ [数对 (i, j) 存在], 那么答案显然是 $\sum_p \prod_{i=1}^{n} a_{ip_i}$

注意到这个式子形式和行列式非常像（其实就是积和式），行列式与积和式唯一的不同的就是系数 -1，而模 2 意义下 -1 和 1 是相等的，所以问题可以直接转化为要求行列式。

permutation

如果我们生成一个矩阵 $a_{i,j} = [\text{数对 (i, j) 存在}]$, 那么答案显然是 $\sum_p \prod_{i=1}^{n} a_{ip_i}$

注意到这个式子形式和行列式非常像（其实就是积和式），行列式与积和式唯一的不同的就是系数 -1，而模 2 意义下 -1 和 1 是相等的，所以问题可以直接转化为要求行列式。

求去掉一个位置之后的行列式值，这个可以通过求出逆矩阵得到代数余子式直接解决。

permutation

如果我们生成一个矩阵 $a_{i,j} = [$数对 (i, j) 存在$]$, 那么答案显然是 $\sum_p \prod_{i=1}^{n} a_{ip_i}$

注意到这个式子形式和行列式非常像（其实就是积和式），行列式与积和式唯一的不同的就是系数 -1，而模 2 意义下 -1 和 1 是相等的，所以问题可以直接转化为要求行列式。

求去掉一个位置之后的行列式值，这个可以通过求出逆矩阵得到代数余子式直接解决。

剩下的问题就是如何消元了，由于矩阵元素只有 01，所以可以使用 bitset 加速。复杂度 $O(\frac{n^3}{32})$

Determinant

给定一个 n 行 $n+1$ 列的矩阵 A，问删去每一列之后的行列式值。

$n \le 200, a_{ij} \le 10^9+7$

Determinant

考虑给矩阵添加上最后一行，变成完整的一个方阵，这样原问题就变成了询问最后一行的代数余子式，可以通过求出逆矩阵解决。

Determinant

考虑给矩阵添加上最后一行，变成完整的一个方阵，这样原问题就变成了询问最后一行的代数余子式，可以通过求出逆矩阵解决。

矩阵不可逆怎么办？

Determinant

考虑给矩阵添加上最后一行，变成完整的一个方阵，这样原问题就变成了询问最后一行的代数余子式，可以通过求出逆矩阵解决。

矩阵不可逆怎么办？

如果存在解，随机一次之后没有逆矩阵的概率是 $\frac{1}{|F|}$ 的。如果随机若干次都是没有逆矩阵的话，说明答案全部是 0。

Determinant

考虑给矩阵添加上最后一行，变成完整的一个方阵，这样原问题就变成了询问最后一行的代数余子式，可以通过求出逆矩阵解决。

矩阵不可逆怎么办？

如果存在解，随机一次之后没有逆矩阵的概率是 $\frac{1}{|F|}$ 的。如果随机若干次都是没有逆矩阵的话，说明答案全部是 0。

这部分的证明我们留到后面。

Determinant

考虑给矩阵添加上最后一行，变成完整的一个方阵，这样原问题就变成了询问最后一行的代数余子式，可以通过求出逆矩阵解决。

矩阵不可逆怎么办？

如果存在解，随机一次之后没有逆矩阵的概率是 $\frac{1}{|F|}$ 的。如果随机若干次都是没有逆矩阵的话，说明答案全部是 0。

这部分的证明我们留到后面。

另外一种方法是可以通过分治消元解决，即分治之后，左边的所有元消全局，然后递归右边，右边的元消全局，然后递归左边

Matrix-Tree 定理

对于无向图 G，定义 G 的度数矩阵 D 满足：

$$d_{ij} = \begin{cases} deg_i & i = j \\ 0 & i \neq j \end{cases}$$

Matrix-Tree 定理

对于无向图 G，定义 G 的度数矩阵 D 满足：

$$d_{ij} = \begin{cases} deg_i & i = j \\ 0 & i \neq j \end{cases}$$

定义 G 的邻接矩阵 C 满足：

$$c_{ij} = \begin{cases} 0 & i = j \\ adj_{ij} & i \neq j \end{cases}$$

Matrix-Tree 定理

对于无向图 G，定义 G 的度数矩阵 D 满足：

$$d_{ij} = \begin{cases} deg_i & i = j \\ 0 & i \neq j \end{cases}$$

定义 G 的邻接矩阵 C 满足：

$$c_{ij} = \begin{cases} 0 & i = j \\ adj_{ij} & i \neq j \end{cases}$$

定义 G 的基尔霍夫矩阵 Q=D-C, 则图 G 的生成树数量是 Q 的任意一个代数余子式。

一些拓展

上面的内容是大家都会的，下面讲一点不是那么普及的东西。

一些拓展

上面的内容是大家都会的，下面讲一点不是那么普及的东西。

带权图的生成树：

定义一棵生成树的权值是所有边的权值。

一些拓展

上面的内容是大家都会的，下面讲一点不是那么普及的东西。

带权图的生成树：

定义一棵生成树的权值是所有边的权值。

我们将度数矩阵中的度数改为边权和，邻接矩阵连通性改为边权，就可以求出来所有生成树的权值和。

一些拓展

上面的内容是大家都会的，下面讲一点不是那么普及的东西。

带权图的生成树：

定义一棵生成树的权值是所有边的权值。

我们将度数矩阵中的度数改为边权和，邻接矩阵连通性改为边权，就可以求出来所有生成树的权值和。

有向图的生成树形图计数：

我们将度数矩阵改为入度，邻接矩阵改为有向边的邻接矩阵，以 x 为根的树形图个数是 C_{xx}

Best theorem

一张有向图 G，欧拉回路的数量是 $t_1(G) * \prod_{i=1}^{n} (deg(i) - 1)!$。

Best theorem

一张有向图 G，欧拉回路的数量是 $t_1(G) * \prod_{i=1}^{n}(deg(i) - 1)!$。

其中 $t_1(G)$ 表示以 1 为根的生成树的数量。

白金元首与独舞

给定一个 n 行 n 列的网格，每个格子有一个向上下左右的箭头或未填。

给每个未填的格子填入一个符号，使得从每个位置出发都能在有限步内走出网格

$n, m \leq 200$, 未填的格子数量 ≤ 300

白金元首与独舞

直接把箭头反向，合法的网格就是一个以外部为根的树形图。

白金元首与独舞

直接把箭头反向，合法的网格就是一个以外部为根的树形图。

直接做树形图计数点数过大，由于未填的很少，可以先把已填好的锁起来，过程中如果有环直接无解，然后对缩起来的图做树形图计数。

白金元首与独舞

直接把箭头反向，合法的网格就是一个以外部为根的树形图。

直接做树形图计数点数过大，由于未填的很少，可以先把已填好的锁起来，过程中如果有环直接无解，然后对缩起来的图做树形图计数。

时间复杂度 $O(nm+k^3)$

更多拓展

Matrix-tree 中用到的性质没有超过环的一般性质，所以任意环的边权都是成立的。这说明我们可以通过一些特殊的边权来做更多的事情，比如多项式环等。

生成树

给定一张 n 个点，m 条边的无向图，每条边有红绿蓝三种颜色

要求绿边数量不超过 g, 蓝边不超过 b 的生成树数量，答案对 10^9+7 取模

$n \le 40$

生成树

考虑给每条绿边一个权值变量 x, 每条蓝边一个权值变量 y，根据之前提到的带权生成树的权值，最终我们看 xy 的次数，就可以知道使用了多少次绿边多少次蓝边

生成树

考虑给每条绿边一个权值变量 x, 每条蓝边一个权值变量 y，根据之前提到的带权生成树的权值，最终我们看 xy 的次数，就可以知道使用了多少次绿边多少次蓝边

问题在于如何求这样一个行列式，我们可以考虑带入 n^2 对点值给 x 和 y，最终使用这些点值进行二维插值，恢复出本来的多项式

生成树

考虑给每条绿边一个权值变量 x, 每条蓝边一个权值变量 y，根据之前提到的带权生成树的权值，最终我们看 xy 的次数，就可以知道使用了多少次绿边多少次蓝边

问题在于如何求这样一个行列式，我们可以考虑带入 n^2 对点值给 x 和 y，最终使用这些点值进行二维插值，恢复出本来的多项式

复杂度 $O(n^5 + n^4)$

生成树计数

给定一个带权无向图，定义一棵生成树的权值是边权和的 k 次方，求所有生成树的权值和，答案对 10^9+7 取模

$n, k \leq 50$

生成树计数

考虑一堆数的 k 次方等价于从这堆数里选出 k 个可重复的数字的积(考虑顺序)，再求和。

生成树计数

考虑一堆数的 k 次方等价于从这堆数里选出 k 个可重复的数字的积 (考虑顺序)，再求和。

那么我们可以将每条边的边权重新定义一个多项式 $\sum_{i=0}^{k} \frac{(vx)^i}{i!}$，这个多项式就表示一条边所代表的权值具体被选了若干次的权值。

生成树计数

考虑一堆数的 k 次方等价于从这堆数里选出 k 个可重复的数字的积 (考虑顺序)，再求和。

那么我们可以将每条边的边权重新定义一个多项式 $\sum_{i=0}^{k} \frac{(vx)^i}{i!}$，这个多项式就表示一条边所代表的权值具体被选了若干次的权值。

对新的权值使用 Matrix-tree, 得到一个 nk 次的多项式，x^k 的系数乘上 $k!$ 就是答案

生成树计数

考虑一堆数的 k 次方等价于从这堆数里选出 k 个可重复的数字的积 (考虑顺序)，再求和。

那么我们可以将每条边的边权重新定义一个多项式 $\sum_{i=0}^{k}\frac{(vx)^i}{i!}$，这个多项式就表示一条边所代表的权值具体被选了若干次的权值。

对新的权值使用 Matrix-tree, 得到一个 nk 次的多项式，x^k 的系数乘上 $k!$ 就是答案

具体求行列式的过程，可以使用上一题的方法，不过只需要一维插值了。

特征多项式

特征多项式

对于方阵 A，如果存在数 λ 和非零 n 阶列向量 x，满足 $Ax = \lambda x$, 则称 λ 是 A 的特征值，x 是关于 λ 的特征向量。

特征多项式

对于方阵 A，如果存在数 λ 和非零 n 阶列向量 x，满足 $Ax = \lambda x$，则称 λ 是 A 的特征值，x 是关于 λ 的特征向量。

考虑如何求解特征值，对原式进行化简，得到 $(A - \lambda I)x = 0$。

特征多项式

对于方阵 A，如果存在数 λ 和非零 n 阶列向量 x，满足 $Ax = \lambda x$，则称 λ 是 A 的特征值，x 是关于 λ 的特征向量。

考虑如何求解特征值，对原式进行化简，得到 $(A - \lambda I)x = 0$。

根据 Cramer 定理，n 个未知数 n 个方程的齐次线性方程组存在非平凡解的条件是系数矩阵行列式为 0。可以得到 $|A - \lambda I| = 0$。

特征多项式

对于方阵 A，如果存在数 λ 和非零 n 阶列向量 x，满足 $Ax = \lambda x$, 则称 λ 是 A 的特征值，x 是关于 λ 的特征向量。

考虑如何求解特征值，对原式进行化简，得到 $(A - \lambda I)x = 0$。

根据 Cramer 定理，n 个未知数 n 个方程的齐次线性方程组存在非平凡解的条件是系数矩阵行列式为 0。可以得到 $|A - \lambda I| = 0$。

该式是一个关于 λ 的一元 n 次方程，称为特征方程，方程左侧称为特征多项式，方程解即为特征值。

特征多项式

特征多项式

通过特征多项式，我们可以优化矩阵快速幂的复杂度。

特征多项式

通过特征多项式，我们可以优化矩阵快速幂的复杂度。

Hamilton-Cayley theorem 说明了，设 M 是域 F 上的矩阵，M 的特征多项式 $P(x)=\sum_{i=0}^{n}a_ix^i$ 是 M 的零化多项式。即 $P(M)=0$。

特征多项式

通过特征多项式，我们可以优化矩阵快速幂的复杂度。

Hamilton-Cayley theorem 说明了，设 M 是域 F 上的矩阵，M 的特征多项式 $P(x) = \sum_{i=0}^{n} a_i x^i$ 是 M 的零化多项式。即 $P(M) = 0$。

这说明了，对于任意的 M^x 都可以由 $I, M, M^2 \ldots M^{n-1}$ 的线性组合表出。

特征多项式

通过特征多项式，我们可以优化矩阵快速幂的复杂度。

Hamilton-Cayley theorem 说明了，设 M 是域 F 上的矩阵，M 的特征多项式 $P(x)=\sum_{i=0}^{n} a_i x^i$ 是 M 的零化多项式。即 $P(M)=0$。

这说明了，对于任意的 M^x 都可以由 $I, M, M^2 \dots M^{n-1}$ 的线性组合表出。

特征多项式

特征多项式

对于 $0 \le x < n$ 结论显然。

特征多项式

对于 $0 \leq x < n$ 结论显然。

考虑归纳证明，对于 x=n，可以直接由定理得到 $a_n M^n = \sum_{i=0}^{n-1} -a_i M^i$。对于 x>n，我们取 $1 \leq y \leq i-1$，$M^x = M^y M^{x-y}$。由于 M^y, M^{x-} 都可以由 $I, M, M^2 \dots M^{n-1}$ 线性表出，那么 M^x 可以由 $I, M, M^2, \dots, M^{2n-2}$ 线性表出。

特征多项式

对于 $0 \le x < n$ 结论显然。

考虑归纳证明，对于 x=n，可以直接由定理得到 $a_n M^n = \sum_{i=0}^{n-1} -a_i M^i$。对于 x>n，我们取 $1 \le y \le i-1$，$M^x = M^y M^{x-y}$。由于 M^y, M^{x-} 都可以由 $I, M, M^2 \dots M^{n-1}$ 线性表出，那么 M^x 可以由 $I, M, M^2, \dots, M^{2n-2}$ 线性表出。

注意到 $M^i P(M) = 0$，可以得到 $a_n M^{n+i} = \sum_{j=1}^{n} -a_j M^{i-j+n}$，我们可以使用这个式子将最高项表示成较低项的线性组合，反复使用这个式子，就可以将 $I, M, M^2, \dots M^{2n-2}$ 的线性组合变换为 $I, M, M^2, \dots, M^{n-1}$ 的线性组合。

特征多项式

特征多项式

如何求特征多项式？

特征多项式

如何求特征多项式？

对于一般的矩阵，只能使用前文提过的方法，代入 n+1 个点值进去得到 n+1 个答案进行插值。

特征多项式

如何求特征多项式?

对于一般的矩阵，只能使用前文提过的方法，代入 n+1 个点值进去得到 n+1 个答案进行插值。

不过一般最常用的情况是常系数齐次线性递推。

常系数齐次线性递推

常系数齐次线性递推

考虑常系数齐次线性递推形成的矩阵。

常系数齐次线性递推

考虑常系数齐次线性递推形成的矩阵。

设数列h_n 满足 k 阶常系数线性递推关系。$h_n = \sum_{i=1}^{k} a_i * h_{n-i}$。

常系数齐次线性递推

考虑常系数齐次线性递推形成的矩阵。

设数列h_n 满足 k 阶常系数线性递推关系。$h_n = \sum_{i=1}^{k} a_i * h_{n-i}$。

写出矩阵后，通过简单的拉普拉斯展开可以发现形成的特征多项式是 $P(x) = x^k - \sum_{i=1}^{k} a_i x^{k-i}$。

常系数齐次线性递推

考虑常系数齐次线性递推形成的矩阵。

设数列h_n 满足 k 阶常系数线性递推关系。$h_n = \sum_{i=1}^{k} a_i * h_{n-i}$。

写出矩阵后，通过简单的拉普拉斯展开可以发现形成的特征多项式是 $P(x) = x^k - \sum_{i=1}^{k} a_i x^{k-i}$。

于是我们可以在 $O(k^2 logn)$ 的时间内完成递推。

常系数齐次线性递推

考虑常系数齐次线性递推形成的矩阵。

设数列h_n 满足 k 阶常系数线性递推关系。$h_n = \sum_{i=1}^{k} a_i * h_{n-i}$。

写出矩阵后，通过简单的拉普拉斯展开可以发现形成的特征多项式是 $P(x) = x^k - \sum_{i=1}^{k} a_i x^{k-i}$。

于是我们可以在 $O(k^2 logn)$ 的时间内完成递推。

更进一步分析，可以发现其实要求的就是 $x^n \mod P(x)$。使用 $O(n\log n)$ 的多项式取模，就可以在 $O(n\log k\log n)$ 的时间内完成递推。

更多拓展

更多拓展

事实上，如果配合 BM 算法，我们可以在更低的时间复杂度内解决稀疏矩阵的问题。暂时不做讨论。

线性空间

线性空间

事实上这部分是之前的知识的基础，只不过因为讲课连续性的需要所以最后才讲。

线性空间

线性空间

线性空间：一个非空集合 V，对加法满足阿贝尔群，对数乘满足结合律，分配律，封闭性，域 F 上的单位元 1 满足 1v=v 。

线性空间

线性空间：一个非空集合 V，对加法满足阿贝尔群，对数乘满足结合律，分配律，封闭性，域 F 上的单位元 1 满足 1v=v 。

子空间：设 W 是 V 的一个子集，W 在加法和数乘下都是封闭的，且 $0 \in W$，则 W 是 V 的子空间。

线性空间

线性空间：一个非空集合 V，对加法满足阿贝尔群，对数乘满足结合律，分配律，封闭性，域 F 上的单位元 1 满足 1v=v 。

子空间：设 W 是 V 的一个子集，W 在加法和数乘下都是封闭的，且 $0 \in W$，则 W 是 V 的子空间。

生成子空间（扩张）：对于若干 V 中的元素 v，包含这些 v 的最小的子空间 W 是这些元素的生成子空间。

线性空间

线性空间：一个非空集合 V，对加法满足阿贝尔群，对数乘满足结合律，分配律，封闭性，域 F 上的单位元 1 满足 1v=v 。

子空间：设 W 是 V 的一个子集，W 在加法和数乘下都是封闭的，且 $0 \in W$，则 W 是 V 的子空间。

生成子空间（扩张）：对于若干 V 中的元素 v，包含这些 v 的最小的子空间 W 是这些元素的生成子空间。

生成集合：对于一个 V 的子集 v，如果 v 的生成子空间是 V，则称 v 是 V 的一个生成集合。

线性空间

线性空间：一个非空集合 V，对加法满足阿贝尔群，对数乘满足结合律，分配律，封闭性，域 F 上的单位元 1 满足 1v=v 。

子空间：设 W 是 V 的一个子集，W 在加法和数乘下都是封闭的，且 $0 \in W$，则 W 是 V 的子空间。

生成子空间（扩张）：对于若干 V 中的元素 v，包含这些 v 的最小的子空间 W 是这些元素的生成子空间。

生成集合：对于一个 V 的子集 v，如果 v 的生成子空间是 V，则称 v 是 V 的一个生成集合。

线性相关

线性相关

对于一个线性空间的一个子集 $v_1, v_2, ..., v_k$，如果 $x_1 v_1 + x_2 v_2 + ... + x_k v_k = 0$ 存在非平凡解，则称这个子集线性相关，否则线性无关。这个条件也等价于任意一个元素可以被其他元素线性表出。

线性相关

对于一个线性空间的一个子集 $v_1, v_2, ..., v_k$，如果 $x_1v_1 + x_2v_2 + ... + x_kv_k = 0$ 存在非平凡解，则称这个子集线性相关，否则线性无关。这个条件也等价于任意一个元素可以被其他元素线性表出。

对于向量空间 V 的一个线性无关的子集 v，如果 v 的生成子空间是 V，则称 v 是 V 的一组基,|v| 是 V 的维度。同时 v 也是 V 最小的生成集合，也是极大线性无关组。

线性相关

对于一个线性空间的一个子集 $v_1, v_2, ..., v_k$，如果 $x_1v_1 + x_2v_2 + ... + x_kv_k = 0$ 存在非平凡解，则称这个子集线性相关，否则线性无关。这个条件也等价于任意一个元素可以被其他元素线性表出。

对于向量空间 V 的一个线性无关的子集 v，如果 v 的生成子空间是 V，则称 v 是 V 的一组基,|v| 是 V 的维度。同时 v 也是 V 最小的生成集合，也是极大线性无关组。

对于一个矩阵 A，如果将 A 的每一行看成一个行向量，那么 A 的极大线性无关组大小称为 A 的行秩，同理可以定义 A 的列秩。可以证明对于同一个矩阵，行秩与列秩是相等的。特别的，如果一个矩阵的秩等于他的阶，则称这个矩阵满秩。

线性相关

线性相关

这里同样可以证明，一个矩阵可逆的条件等价于矩阵满秩。

线性相关

这里同样可以证明，一个矩阵可逆的条件等价于矩阵满秩。

如果不满秩，则存在一个线性相关组，那么在消到最后一行的时候，由于可以被之前的向量线性表出，所以一定被消成 0 了。所以一定不可逆。

线性相关

这里同样可以证明，一个矩阵可逆的条件等价于矩阵满秩。

如果不满秩，则存在一个线性相关组，那么在消到最后一行的时候，由于可以被之前的向量线性表出，所以一定被消成 0 了。所以一定不可逆。

同样我们可以证明 Determinant 一题中的结论。

考虑 n 个线性无关的向量，张成的空间大小是 $|F|^n$ 的。而 n+1 维向量的空间总大小是 $|F|^{n+1}$ 的。因为矩阵不可逆等价于线性相关等价于新加入的元素可以被表出，因为就是新加入的元素在之前已经张成的空间内的概率，显然是 $\frac{|F|^n}{|F|^{n+1}}$ 的。如果多次随机都是不可逆，说明本来的向量就已经有线性相关了，那么删去任意一维都不可能改变。

线性基

其实线性基就是维护的一组线性无关的基。每次加入新的向量的过程，就是一个经过之前的元的消元的过程。

Intersection

给定两个集合 A 和 B，定义一个整数 x 属于某个集合，当且仅当存在一个子集的异或和等于 x。求同时属于两个集合的数字个数。

集合大小 ≤ 50, 集合元素 $\leq 2^{60}$。

Intersection

Intersection

考虑 $span(A) \cap span(B)$ 和 $span(A) \cup span(b)$ 全部是子空间。

Intersection

考虑 $span(A) \cap span(B)$ 和 $span(A) \cup span(b)$ 全部是子空间。

要求 $span(A) \cap span(b)$ 的维度，求出 A 的维度，B 的维度，$span(A) \cup span(b)$ 的维度即可。

矩阵

给定一个 n 行 m 列的 01 矩阵 A。

如果矩阵第 j 列有奇数个 1 ，就有 $a_j * 3^{b_j}$ 的贡献。

现在可以删去任意行，最大化最终的贡献和。

$n \leq 200000, m \leq 70, a_j = \pm 1, b_j \leq 35$。保证同样一对 (a_j, b_j) 只出现一次。

矩阵

矩阵

注意到 aj 的范围只有 3，所以可以从高位向低位贪心，每次贪心的尝试是否可以达到某一位的取值要求。

矩阵

注意到 aj 的范围只有 3，所以可以从高位向低位贪心，每次贪心的尝试是否可以达到某一位的取值要求。

保留任意行这个条件，其实就是选择若干个向量组合起来，所以也就是询问生成空间中是否含有某个特定的向量。

矩阵

注意到 aj 的范围只有 3，所以可以从高位向低位贪心，每次贪心的尝试是否可以达到某一位的取值要求。

保留任意行这个条件，其实就是选择若干个向量组合起来，所以也就是询问生成空间中是否含有某个特定的向量。

这个问题可以直接通过消元解决。复杂度 $O(\frac{nm^3}{32})$。

矩阵

注意到 aj 的范围只有 3，所以可以从高位向低位贪心，每次贪心的尝试是否可以达到某一位的取值要求。

保留任意行这个条件，其实就是选择若干个向量组合起来，所以也就是询问生成空间中是否含有某个特定的向量。

这个问题可以直接通过消元解决。复杂度 $O(\frac{nm^3}{32})$。

注意到满空间的维度不会超过 70，所以预先求找出来一组基即可。复杂度 $O(\frac{nm^2+m^4}{32})$。

最后

祝大家 WC 愉快。