Codechef 13 June SPMATRIX Solution

安师大附中 罗哲正

题目大意

给一个n阶方阵要求满足：

(a) $A[x][x] = 0 for 1 \le x \le N.$

(b) $A[x][y] = A[y][x] > 0 for 1 \le x < y \le N.$

(c) $A[x][y] \le max(A[x][z], A[z][y]) for 1 \le x, y, z \le N.$

(d) $A[x][y] \in 1, 2, \ldots, N-2 for 1 \le x < y \le N.$

任意 $k \in \{1, 2, \ldots, N-2\}$ 存在 $x, y \in 1, 2, \ldots, N$ 满足 $A[x][y] = k$.

答案模10^9+7，$n \le 10^7$

解决方案

发现输入只有一个n，这启发我们使用OEIS 或者打表，可以对小数据暴力，然后得到前几个数的表输入OEIS。不过还有一种办法，观察样例，给了一个$n=10$的情况，由于数列指数级增长，该数字应该是这个序列独一无二的，答案模了$M = 10^9+7$假设$ans = Mq + r$，我们可以枚举q，十次以内就可以找到ans，从而在OEIS 上找到该序列http://oeis.org/A059355。

公式：

$$ans = \frac{N!(N-1)!}{2^{N-1}} \left(\frac{N}{2} - \frac{2}{3} - \frac{H_{N-1}}{3} \right)$$

其中$H_n = \sum_{k=1}^{n} \frac{1}{k}$

于是预处理阶乘和阶乘逆元，每次就可以做到$O(1)$了。

证明

你要寻找的矩阵和距离矩阵非常类似(满足类似的三角不等式)。实际上，满足(a)-(c)的矩阵被称为超度量空间(可以理解为图的某种最短距离矩阵)，而X还可以代表图的点集。

对于一个超度量空间X，定义X的谱$Sp(X)$表示X中非0元素的集合。那么给定一个谱S，我们可以把它变换为一个等价的谱 $\{1, 2, \ldots, |S|\}$ ，保证元素间相对顺序不变，类似于离散化，我们也可以变换回来。那么我们要解决的问题就是，给定N，求满足 $|X| = N$且$|Sp(X)| = N-2$ 的超度量空间X的个数。

我们对于一个超度量空间X，可以定义$diam(X)$表示X的直径，指的是图中最远的两个顶点的距离。现在让我们寻找一些超度量空间的有用的属性。接下来$M[x][y]$也可以用$d(x, y)$或者$dist(x, y)$表示。

直观的来看，我们应该以特定的顺序把值放到$dist(x, y)$中。例如升序(从1到$diam(X)$)，或者是降序(从$diam(X)$到1)，最终我们来考虑降序填写。

考虑一个建立在点集上的二元关系 $R_d, d > 0$ ，定义 $R_d(x, y)$ 为 $dist(x, y) < d$。 显然R_d是一个定价关系，证明很简单略去。

进一步我们可以得出，如果我们按照降序来填写$d(x,y)$，由于X是一个满足三角不等式的矩阵，我们将不断的按照等价关系R_d分割X的点集，那么分割集合X之后跨越不同等价类的边都满足$d(x,y)=diam(X)$。

根据上面的结论我们可以发现 $|Sp(X)|\le|X|-1$ ，因为集合最多划分$|X|-1$层。也可以通过归纳法证明，很简单因而这里不做详细说明。

我们需要注意的是对于任意超度量空间X，$|Sp(X)|=|X|-1$始终是满足的，当我们将X分成两部分的时候，两个部分的谱是独立的。

设$f(N,K)$表示满足 $|X|=N$且$|Sp(X)|=K$ 的超度量空间X 的数目。则我们的目标就是求出$f(N,N-1)$。

现在假设我们已经有了一个X，而现在我们需要让$|Sp(X)|=|X|-2$，一共有三种情况：

(A1)按照等价关系R_{N-2}，X被分成了3个等价类，每个等价类的谱不相交。

(A2)按照等价关系R_{N-2}，X被分成了2个等价类，两个等价类的谱只有一个元素相交。

(A3)按照等价关系R_{N-2}，X被分成了2个等价类，两个等价类谱不相交，但是有一个部分X_i满足$|Sp(X_i)|=|X_i|-2$。

我们现在可以分别对这三部分进行递推计算，不过我们首先发现$f(N,N-1)$出现频率很高，所以先考虑计算$f(N,N-1)$。

很显然每次X只能被划分成恰好两个不相交的独立的等价类，枚举元素划分和谱的划分。容易得到递推式：
$f(N,N-1)=\sum[1\le k\le N-1]\binom{N-1}{k-1}*\binom{N-2}{k-1}*f(k,k-1)*f(N-k,N-k-1)$ ，显然$f(k,k-1)$和$f(N-k,N-k-1)$都是$f(N,N-1)$相同的规模跟小的问题。

通过归纳法，我们可以证明$f(N,N-1)$的公式：$f(N,N-1)=\frac{N!(N-1)!}{2^{N-1}}$。 不过有更加直观的组合意义的理解：

首先我们把所有点和谱值间隔排列(共N个点，$N-1$个谱值)，这样一共有$N!(N-1)!$种排法。然后利用这个排列进行等价类划分，首先我们找到最大的谱值，将排列从此处断开，分成两边两个等价类，两边递归划分。也可以理解为将谱值做笛卡尔树，把点看成笛卡尔树的叶子。这样显然一个排列对应一种划分方法，而一个划分方法会一共划分$N-1$次，每次划分的两个等价类可以交换，所以一个划分方法共被计算了2^{N-1}，所以除以2^{N-1}就能得到：$f(N,N-1)=\frac{N!(N-1)!}{2^{N-1}}$。

剩下的便是计算A1，A2，A3。

(A1)按照等价关系R_{N-2}，X被分成了3个等价类，每个等价类的谱不相交。

通过枚举的这三个等价类之间的关系，我们可以得到方案数的公式：

$$\sum[n_1+n_2+n_3=N][n_1,n_2,n_3\ge 1]\binom{N-1}{n_1-1}*\binom{N-n_1-1}{n2-1}*\binom{N-3}{n_1-1}*\binom{N-n_1-2}{n_2-1}$$
$$*f(n_1,n_1-1)*f(n_2,n_2-1)*f(n_3,n_3-1)$$

我们依旧采用计算$f(N,N-1)$的方法，考虑N个点中插入$N-1$个谱值，并做类似笛卡尔树，不同的是，这次我们让根节点有三个孩子，分别以最大和次大值作为分界。则排列数依旧是$N!(N-1)!$，共进行了$N-3$次分割成两部分的操作，一次分割成三部分的操作，所以要除以$2^{N-3}*3!$，最大值和次大值可以交换，所以再除以2，我们得到最终的公式是$A1=\frac{N!(N-1)!}{2^{N-1}*3}$。

(A2)按照等价关系R_{N-2}，X被分成了2个等价类，两个等价类的谱只有一个元素相交。

我们把X分成大小分别为k和$N-k$的等价类，两个等价类的谱有一个公共元素，枚举k和这个公共元素我们可以得到式子：

$$\sum[2\le k\le N-2]\binom{N-1}{k-1}*\binom{N-3}{k-1}*(k-1)*f(k,k-1)*f(N-k,N-k-1)$$

我们还是采用那个方法：我们从1到$N-3$任取一个值k在谱值排列中出现两次，如果我们把这两个k区分开来且强制一个在；另一个前面，排列总数是$\frac{N!(N-1)!}{2}$种，还要满足$N-2$在两个k之间，由于$N-2$和两个k的相对位置有3中情况所以还要除以3。

于是我们可以得到A2的计算公式：$A2=(N-3)*\frac{N!(N-1)!}{6*2^{N-1}}$

(A3)按照等价关系R_{N-2}，X被分成了2个等价类，两个等价类谱不相交，但是有一个部分X_i满足$Sp(X_i)=|X_i|-2$。

依旧先考虑暴力枚举等价类划分大小k和$N-k$，并规定谱的大小为$k-2$和$N-k-1$。

可以列出算式：$\sum[3\le k\le N-1]\binom{N}{k}*\binom{N-3}{k-2}*f(k,k-2)*f(N-k,N-k-1)$

不幸的是，使用枚举排列的方法很难计算方案重复的次数，我们只能寻找其他办法。

我们将A3的式子展开，只保留$f(k,k-2)$，可以得到：

$$A3=\frac{N!*(N-3)!}{2^{N-1}}*\sum[3\le k\le N-1]\frac{2^k}{k!(k-2)!}*f(k,k-2)$$

令$g(N)=f(N,N-2)*\frac{2^{N-1}}{N!(N-1)!}$，接着我们直接计算$g$。

$$g(N)=\frac{N-1}{6}+\frac{2}{(N-1)(N-2)}*\sum[3\le k\le N-1](k-1)g(k)$$

我们考虑一般求通项的方法，考虑$g(N+1)$和$g(N)$

$$g(N+1)=\frac{N}{6}+\frac{2}{N(N-1)}*\sum[3\le k\le N](k-1)g(k)$$

$$g(N)=\frac{N-1}{6}+\frac{2}{(N-1)(N-2)}*\sum[3\le k\le N-1](k-1)g(k)$$

将$g(N+1)$乘N，$g(N)$乘$N-2$，做差。

$$Ng(N+1)-(N-2)g(N)=\frac{3N-2}{6}+2g(N)$$

于是$g(N+1)=g(N)+\frac{1}{2}-\frac{1}{3N}$

而我们已经知道的是$g(3)=\frac{1}{3}$

很容易的可以推出通项：$g(N)=\frac{N}{2}-\frac{2}{3}-\frac{H_{N-1}}{3}$，其中$H_n$是调和数($H_n=\sum_{k=1}^{n}\frac{1}{k}$)。

计算最终答案$f(N,N-2)=\frac{N!(N-1)!}{2^{N-1}}*g(N)=\frac{N!(N-1)!}{2^{N-1}}\left(\frac{N}{2}-\frac{2}{3}-\frac{H_{N-1}}{3}\right)$

撒花~

参考资料

Codechef官方题解 https://discuss.codechef.com/questions/13633/spmatrix-editorial