不问天（match）解析

给出一个 n 阶图 G，对于 $k=1,\dots,l$ 问 k 个 G 放在一起然后取反得到的图有多少种匹配。

Intended Solution: $\Theta(n^2(l+2^{n/2}))$。

part1

首先考虑容斥，我们设 e_k 是从 G 中选出 k 条边的匹配的方案数。那么我们考虑这样一个多项式：

$$E(x)=\sum_k e_k(-1)^k x^{n-2k}$$

这就是说选一条边就乘以 -1 的代价。我们知道 k 个 G 放在一起会得到 E^k。

考虑 $I_{k,r}$ 为 $E(x)^k x^r$ 对应的答案（接下来记为 $D(E(x)^k x^r)$），那么对于 $r>0$，考虑先强制选一个 x，然后要么这个点不匹配，要么从 x^{r-1} 里选一个，要么和 $E(x)^k$ 里选一个，因此有 $D(E(x)^k x^r)=D(kE(x)^{k-1}E'(x)x^{r-1}+E(x)^k((r-1)x^{r-2}+x^{r-1}))$，也就有

$$I_{k,r}=\left(k\sum_j I_{k-1,r-1+j}[x^j]E'\right)+(r-1)I_{k,r-2}+I_{k,r-1} \quad (1)$$

注意，又有转化关系

$$I_{k,r}=\sum_j I_{k-1,r+j}[x^j]E \quad (2)$$

因此我们可以只计算 $0\le k\le l, 0\le r\le 2n-1$ 的部分。

我们来明确一下运算顺序：首先算出 $I_{0,\cdot}$，然后对于 $k=1,\dots,l$ 从小到大，再令 r 从 0 到 n，可以通过 (1) 式子求出 $I_{k,r}$，然后通过 (2) 式子求出 $I_{k-1,r+n}$（用于计算后面的 r！）。

这一部分直接递推的复杂度是 $\Theta(n^2 l)$。

（类比**棋盘多项式（Rook Polynomial）**，我们是不是可以将刚刚用到的计算叫做匹配多项式呢？）

part2

在这之前我们需要算一下 $E(x)$，也就是 e_k。

> 我们先不妨将 n 补成偶数。考虑全体边 $(2i-1,2i)$ 构成的匹配 M_2，对于任何一个匹配 M_1，$M_1\cup M_2$ 的形态是若干个环和路径。这其实就有点像是把 $2i-1$ 和 $2i$ 缩成了一个点，然后我们可以对这 $2^{n/2}$ 个点上面跑一个 $\Theta(n^2 2^{n/2})$ 的状压 DP。那么对于每个集合 $\varnothing\ne S\subseteq\{1,\dots,n/2\}$，我们设 f_S 和 g_S 表示 S 被路径覆盖和环覆盖的方案数，而路径就相当于是少匹配一条边。

原题的 idea 差不多就到这里了。接下来的 n 是指原来的 $n/2$。

我们考虑计算的就是对于所有集合划分，将其 $\prod(f_S+g_S t)$ 求和。那么最后结果中的 t^k 次项系数就是 $n/2-k$ 条匹配边的方案。这传统做法是 $\Theta(n^3 2^n)$，但是事实上可以 $\Theta(n^2 2^n)$。

首先考虑 $h(S)=\sum_{T\supseteq S} f(T)g(T\setminus S)$，这个使用子集卷积类似的处理方式即可做到 $\Theta(n^2 2^n)$。现在我们设 $P_k(S,t)$ 表示一个集合划分中有数 $>k$ 的集合去掉之后，剩下的集合构成 S 的划分，此时有数 $>k$ 的那些集合的 $\prod(f_S+g_S t)$ 的乘积之和。注意由于有数 $>k$ 的集合最多只有 $n-k$ 个，也就是说多项式的次数是 $n-k$ 的。接下来我们考虑 k 从 n 递减到 0 的维护过程，无非是将 $S\ni k$ 的集合 S 都去掉，这也就是刚刚所提到的那种卷积。考虑多项式的次数，这里的问题规模之和是 $\sum_{k\le n} 2^k(n-k+1)=\Theta(2^n)$，因此复杂度还是 $\Theta(n^2 2^n)$。