好题分享

李可

长郡中学

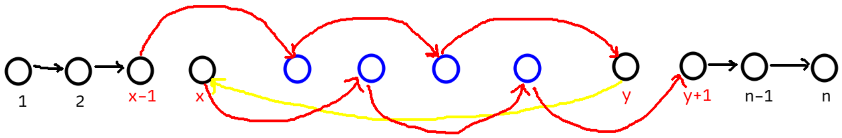
2025 年 4 月 27 日

CF1616G Just Add an Edge

给定一个特殊的 DAG，有 n 个点，m 条边，满足每条边都是 $x \to y, x < y$ 的形式。这个图一开始可能并不存在哈密顿路径，所以你可以添加一条边来使得它存在哈密顿路径。
具体地说，请你求出有多少条本质不同的边 $u \to v, u > v$ 满足加上边 $u \to v$ 之后图中存在了哈密顿路径，如果添加前就存在哈密顿路径，添加这条边仍然是可以的。
$1 \leq n, m \leq 5 \times 10^5, 1 \leq x, y \leq n$。

CF1616G Just Add an Edge

首先如果原图有哈密顿路径，那肯定是 $1 \to 2 \to 3 \to \cdots \to n-1 \to n$ 这样的形式，此时添加任意的边都可以，答案就是 $\frac{n(n-1)}{2}$。
现在假如我们加了一条边，是 $y \to x, x < y$，那么最终的哈密顿路径的形式一定是这样的：



CF1616G Just Add an Edge

我们容易想到对合法的 $(x-1,x),(y,y+1)$ 对计数，考虑枚举 $(x-1,x)$，然后数它右边有多少个合法的 $(y,y+1)$。这样的 $(x-1,x),(y,y+1)$ 需要满足这些条件：

- 存在 $1\xrightarrow{+1\text{ each}}x-1,y+1\xrightarrow{+1\text{ each}}n$ 的路径。
- 存在 $x-1\rightsquigarrow y,x\rightsquigarrow y+1$ 的路径，且这两条路径不交，经过的点的并集为 $[x-1,y+1]$。

第一个条件是好判断的，我们维护 r_i 表示对于 i 点，最大的 j 满足存在路径 $i\xrightarrow{+1\text{ each}}j$，那么这个条件就相当于 $r_1\geq x-1,r_{y+1}\geq n$。

CF1616G Just Add an Edge

第二个条件看着不好判，但是路径本身性质很强大，点集并连续无遗漏，并且点集没有交，我们不妨对这个路径进行拆分：
假设一开始的位置用有序点对 $(x-1,x)$ 表示，我们每次让两个点分别往后跳一定的距离，假如 $x-1$ 跳到了一个 $v\in e(x-1), v>x$，此时我们就会发现，$[x,v-1]$ 范围内的点都得给 x 跳完，所以当 $x-1\to v$ 时，$x\xrightarrow{+1\text{ each}} v-1$。这时候位置的有序点对变成了 $(v,v-1)$。
假如每次我们都考虑其中一个点往前跳一条非 $i\to i+1$ 形式的边，那么下一次得到的点对也会是 $(i,i+1)/(i+1,i)$ 形式的。而初始状态为 $(x-1,x)$，所以可以归纳得到，任何时候位置点对都是 $(i,i+1)/(i+1,i)$ 形式。然后这个点对 $(u,u+1)$ 一次的移动应当满足
$u\to v+1, u+1\xrightarrow{+1\text{ each}} v$，然后就有 $(u,u+1)\to(v+1,v)$。

CF1616G Just Add an Edge

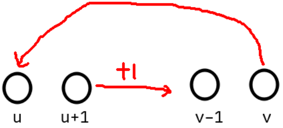
这个过程我们考虑用 dp 来记录可行性，为了格式的统一，我们把一开始的 $(x-1,x)$ 变成 $(x,x+1)$。$f_{i,0/1}$ 表示从一开始的 $(x,x+1)$ 能否跳到 $(i,i+1)/(i+1,i)$ 的位置点对。转移是简单的，考虑 $(u,u+1)\to(v,v-1)$，即满足 $u\to v,u+1\xrightarrow{+1\text{ each}}v-1$，也就是 $v\in e(u),r_{u+1}\ge v-1$，那么就有 $f_{v-1,t}\leftarrow f_{u,1-t}$。
对于每个开头 $(x,x+1)$，我们可以 $O(n)$ 的 dp 求出它右侧合法的 $(y,y+1)$ 个数。于是这样我们就能 $O\left(n^2\right)$ 做了，因为是 $0/1$ 的 dp 值，可以 bitset 优化做到 $O\left(\frac{n^2}{\omega}\right)$。

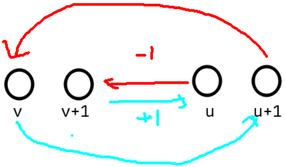
CF1616G Just Add an Edge

考虑优化，那就一定不能再枚举 $(x, x+1)$ 了。我们不妨进一步观察图的性质，发现如果将 $p \nrightarrow p+1$ 的 p 称为断点，假设 L, R 分别为第一个，最后一个断点，那么 $(x, x+1), (y, y+1)$ 中，应满足
$x \in [1, L], y+1 \in [R, n]$，而中间的路径是 $(L, L+1)$ 往后的路径或者后面往前走到 $(L, L+1)$ 的路径。
这时候我们发现，将中间 $[x, y+1]$ 的路径 $x \rightsquigarrow y, x+1 \rightsquigarrow y+1$ 看成，从 $(x, x+1) \rightsquigarrow (y, y+1)$ 的路径。

CF1616G Just Add an Edge

这时候我们就能够直接 dp 了，我们发现 $\geq L$ 的部分可以直接从 $(L, L+1)$ 开始进行之前的 dp，但是 $\leq L$ 的部分我们不能直接从 $(L, L+1)$ 开始，因为相当于是反向的，是从 $(x, x+1) \to (L, L+1)$。这时候我们不妨考虑建一个反图，然后对于 $[1, L]$ 的部分，我们从后往前按照反图的边来 dp。两个方向的转移分别如图：





CF1616G Just Add an Edge

那么最后我们计算合法的 $(x, x+1), (y, y+1)$ 组数的时候，可以将 x, y 分开统计再乘法原理。那么我们统计出满足 $x \in [1, L], f_{x,0} = 1$ 的个数 cx_0， $y+1 \in [L, n], f_{y,0} = 1$ 的个数 cy_0，贡献了 $cx_0 \times cy_0$ 个点对。同理 $cx_1 \times cy_1$ 的贡献也能一样的计算。然后
$x \in [1, L], f_{x,0} = f_{x,1} = 1, y+1 \in [L, n], f_{y,0} = f_{y,1} = 1$ 的贡献会被算两次，减掉这里的贡献就好了。
有一些小 corner：

- dp 的时候初始状态不好定义，因为可能出现 $L = 1$ 或者 $R = n$ 这一类的情况，这时候难以钦定路径的起终点，于是我们不妨建虚点 $0 \to (1, n], [1, n) \to n+1$。
- 如果只有一个断点 p，那么统计的时候会把 $p \to p+1$ 也统计进去，但是显然是不合法的，所以要 -1。

复杂度 $O(n)$。

P9257 [PA 2022] Mędrcy

现在有 n 个人，m 条咒语，一开始的时候，每条咒语恰好被 $n-2$ 个人知道。每个人了解自己知道的咒语相关的所有信息，也就是说他知道自己知道的每条咒语，被哪些人知道了，但是他不知道任何自己不知道的咒语相关的信息。人之间也不会交流关于自己知道的咒语的信息。

初始时是第 0 天，此时现在大家都知道了一件事，也就是至少存在一条咒语。在第 1 天到第 k 天的过程中，如果当天一个人发现肯定存在一条咒语他不知道，这个人就会在第二天的时候消失。人可以通过每天消失的人的信息和天数推断出自己是否有不知道的咒语。

现在给出 m 条咒语以及它被哪些人知道。问第一次有人消失的日子是否 $\leq k$，如果是，则输出这一天，以及在这一天消失的人的集合。

$3 \leq n, 1 \leq m, 1 \leq k \leq 30, 1 \leq a_i < b_i \leq n,$

多测，一组数据中所有测试点的 n 之和不超过 10^3，所有 m 之和不超过 3×10^3。

P9257 [PA 2022] Mędrcy

首先考虑一个人会在第 i 天消失的条件是什么。

- 如果人 x 在第 1 天消失，说明 x 不知道任何一个预言。
- 如果人 x 在第 2 天消失，说明他不能在第 1 天消失，然后存在一个人 y，满足他也没在第 1 天消失，并且不存在一条预言，x, y 同时知道。
- 如果人 x 在第 3 天消失，说明他不能在第 $1, 2$ 天消失，然后存在两个没在第 $1, 2$ 天消失的 $y, z,$，不存在一条预言，x, y, z 同时知道。
- 由此，我们可以归纳出人 x 在第 i 天消失的条件，就是存在一个集合 S，满足 $x \in S, |S| = i, S$ 中的人都还没消失，然后不存在一条预言满足 S 内的人都知道。

P9257 [PA 2022] Mędrcy

这时候我们发现，如果考虑一个集合 S 是否都知道一个预言是不好做的，因为预言的特殊性质是恰好有两个人不知道。我们不妨转化条件为，对于每个预言 j，假设不知道它的两个人分别为 a_j, b_j，那么 S 满足包含 a_j, b_j 中至少一个。
两个中覆盖至少一个，想到最小点覆盖。
我们将人看作点，将每个 a_j, b_j 之间连一条无向边，那么相当于 S 就是最小点覆盖，因为有人消失的最早天数 i 就是 $|S|$ 的大小。

P9257 [PA 2022] Mędrcy

梳理一下现在的问题，相当于对于一个 n 点 m 边的无向图，我们要求出它的最小点覆盖大小，以及所有可能出现在最小点覆盖中的点的集合。因为要求可能出现的点，我们只能够想办法求出所有的最小点覆盖方案。这里有一个条件性质，$k \le 30$，而如果最小点覆盖比 k 大，就不用求了。于是我们考虑暴搜一类的做法，问题在于我们希望指数在 k 上而不是 n 上。这里有一个简单的想法，因为是需要覆盖所有边，所以对于任意一个点 u，如果不选点 u，就得选择 $e(u)$ 即 u 的邻集中所有点。

P9257 [PA 2022] Mędrcy

我们考虑顺着决策每个点是选择它还是选它的邻集，这样我们最多决策 k 个点。每个点的选择是 2 种，每个点选完之后需要用 $O(n)$ 的时间更新这个选择的影响，所以复杂度 $O\left(2^k n\right)$。
这个做法已经很有前途了，我们考虑对它进行一些优化。暴搜的优化无非这几种思路：

- 优化搜索顺序
- 去除重复的，不优的搜索内容
- 在搜索到一定程度时，剩下的内容有更好的性质，就用非搜索的复杂度更优的做法解决

P9257 [PA 2022] Mędrcy

- 首先考虑优化搜索顺序，我们不妨每次搜剩余度数最大的点（邻集大小最大的），这样每次如果选了邻集，还能选的点数会更少
- 去除重复不必的搜索内容，在搜之前去除重边
- 当剩下的点中度数最大值为 2 时，只剩下了一些简单环和链，那么剩下的方案已经确定了，我们每次搜一个环/链出来，根据其长度奇偶性确定答案。

这时候我们保证，搜索的时候，如果不选当前点，就会选择一个至少有 3 个点的集合，假设 $T(k)$ 为还剩 k 个点时的搜索次数可能最大值，就有 $T(k)=T(k-1)+T(k-3)$，可以算出 $T(30)=58425$。
最终时间复杂度为 $O(T(k)n)$。

谢谢大家，祝大家学业有成！