《造数据》解题报告

北京市第八十中学 曹越

一. 题目大意

对于一张n个点，m条边，有标号，无自环无重边无孤点的无向图，定义它的变换是构造方式如下的有向图：

1.有向图有m个点，与无向图m条边一一对应。

2.对于无向图中每个点，将与它相连的边按照另一个端点从小到大排序，并在排序中相邻的两条边在有向图中的对应点从前向后连边。

定义有向图G的一个状态为给每个点分配0,1,2之一的值，称之为出现次数，给每条边分配一个0,1之一的值，称之为这条边走过/未走过。

定义一个状态合法当且仅当对于每个点下面五个条件均不满足：

1.这个点值为0且它的入边和出边值不全为0。

2.这个点值为2且它的入边和出边值不全为1。

3.这个点入度为2，且这个点入边值之和不等于这个点的值。

4.这个点出度为2，且这个点出边值之和不等于这个点的值。

5.这个点的出边所能到达的点的值不全都小于等于这个点的值。

现在请你构造一个点数为n的有向图G，满足下面几个条件：

1.G能被某个有标号无自环无重边无孤点的无向图变换而来。

2.G中的边都看成无向边之后G连通，即G是弱连通的。

3.G的合法状态数尽量的多。

4.在满足 3 的条件下，G的边数尽量小。

每次给定n，输出满足条件的任意一种图以及它的合法状态数。

二. 数据范围

对于所有数据满足$n \leq 400$。

数据范围表格如下：

测试点编号	$n =$	测试点编号	$n =$
1	3	11	75
2	5	12	100
3	7	13	150
4	8	14	200
5	9	15	250
6	10	16	300
7	11	17	325
8	15	18	350
9	20	19	375
10	50	20	400

三. 解题过程

3.0 算法 0

按照样例输出，可以通过第一个测试点。

期望得分：5 分。

3.1 算法 1

首先最简单的思路就是暴力搜索图G，然后计算G中合法状态的总数。可以发现G是一张有向无环图，因此对图G按照拓扑序进行标号之后，每条边一定是由标号较小的点连向较大的点。图G的可能情况大概是$O((n!)^2)$级别的。

计算状态总数时可以采用枚举每条边是否经过，再求出每个点的可能的经过次数。这样做时间复杂度为$O(2^m)$，大约可以很快的搜索出$n = 7$以内的情况。

期望得分：15 分。

3.2 算法 2

进一步对图G的合法状态进行分析。

首先小 C 所说的五个条件有些繁琐，问题主要是在出入度是否为2的讨论上。由于在无向图的变换过程中，每条链的开头和结尾只增加入度或只增加出度，不满足出入度平衡，因此我们可以将所有入度不为2的点补充1或2条没有起点的入边，出度也相应的补充，再把无向图中每个点对应的链添加开头和结尾两条补充的边，那么G中每一个点出入度都是2了。这里需要注意的是入度/出度为0的情况，两条补充的边应看成相同的。

不难发现这个时候条件就转化为了：

1. 每个点的经过的出入边数平衡。
2. 拓扑序靠前的点经过次数不比靠后的点更小。

这个有什么用呢？通过算法 1 打表或者其他各种方式，可以看出如果把图G的边变成无向边，G应该是一棵树。其实通过小 C 的初始条件应该也可以证明，但是经过这一步转化后证明更简洁：

对于图G看成无向图后存在环的情况，找出一个环，这个环上一定存在一个点使得环上与这个点相邻的边都是这个点的出边。显然如果找不到那么沿着环的方向拓扑序是单调的，但是G是有向无环图，所以一定找得到。

接下来将这个点的两条出边其中一条断掉，形成一条没有起点的边和一条没有终点的边。对于原来图中的每一种方案，考虑新图中这样的方案：令断掉的这两条边是否经过的状态都和原图相同，其他边的状态不变。这样构造出来新图的方案是合法的，而且不会重复。

由此就证明了图G看成无向图后是一棵树比不是一棵树不会更劣，由于要求输出的答案边数最小，因此输出的结果一定是一棵树，否则就是错误的。

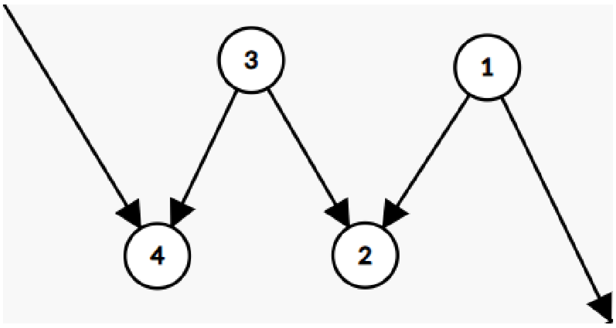
这样可以优化算法 1 中第一步的暴力搜索过程，按照拓扑序标号后，搜索带标号无根树，根据 prufer 序列，可能的树总数最多为$O(n^{n-2})$。容易较快的搜索出$n = 9$之内的答案。

期望得分：25 分。

3.3 算法 3

图G已经变成了树状，考虑如何快速计算合法状态数。

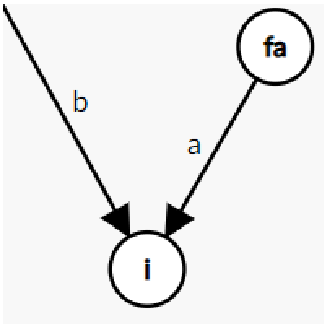
继续观察合法状态的情况，不难发现图G中存在下面这种 w 状的链结构：



把这条链上面的边按顺序编号后，可以看出：这条链上面奇数位置的边和偶数位置的边必须分别按顺序经过。例如图中的$1-2$与$3-4$，如果$1-2$经过了但是$3-4$没经过，那么2号点经过次数就是$[3-2\text{经过了}]+1$，而3号点经过次数为$[3-2\text{经过了}]$，不符合条件。用类似的方式，就可以证明奇数位置的边和偶数位置的边是按一定顺序经过的，后面的边走过了前面的边也一定要走过。

这样的 w 状链还有一个好处，就是链上的边状态确定以后，可以直接确定链上面的点的经过次数。这样合法的状态数可以转化为所有 w 状链的状态，满足每个点在它所在的所有 w 状链上出现次数相同。

由此引入一个树形 dp，记$f_{i,a,b}$表示第i个点的子树，i的父节点向i的边是否走过，以及与这条边在 w 状链上相邻的边是否走过。如图：



其中fa表示i的父节点。对于反方向的情况，即i连向i的父节点的情况，并不需要单独记录，可以发现将一个子树内的边是否经过的状态取反，再把每条边的方向取反，这时仍然是一个合法的方案。因此在转移的时候只需要枚举最多三个子节点6条边的状态。dp 一次的时间复杂度为$O(n)$。暴力搜索复杂度降为$O(n^{n-1})$，可以搜索出大约$n=11$以内的答案。

期望得分：35 分。

3.4 算法 4

有了算法 3 的树形 dp 做法，一种自然的思路就是在搜索的过程中将一定不优的方案去掉。具体做法就是从1到n进行递推，对于i，枚举三个子树，他们的大小之和为$i-1$，并由此求出所有大小为i的子树的可能形态，以及它们的f值，如果一种形态的f值都小于等于另一种，那么这种形态就一定不会更优了，删掉就可以。值得注意的是算法 3 中的 b 边不经过一定是要比经过更优的，所以在比较的时候可以把四个维度改为$f_{i,0,0}, f_{i,0,0}+f_{i,0,1}, f_{i,1,0}, f_{i,1,0}+f_{i,1,1}$，这样状态数还会更少。

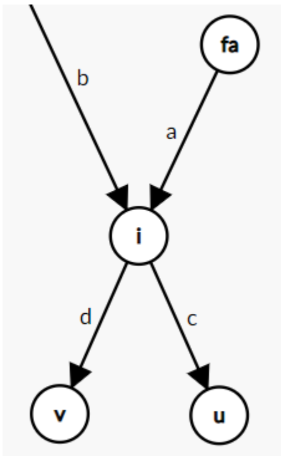
由于这个做法需要枚举三个子树，因此复杂度大约为1到n所有可能形态总数的三次方。

经过实际操作，可以发现当n达到60左右时，对于每个i状态数依旧在100左右，但答案已经超过了 long long 范围，需要手写一个简单的高精度。但由于高精度带来的常数，大约也只能得到$n=150$左右的答案。

期望得分：65 分。

3.5 算法 5

枚举三个子树会花费大量的时间，两个 w 状链之间边的限制是独立的，只需要让它们在i这个点经过次数相同就可以了。如图：



搜索出大小和为i的u,v的可能形态，这个时候u,v的贡献是g_0,g_1,g_2，表示i的经过次数是0,1,2时u,v两个子树的方案数。这个同样也可以剪枝。算出这些u,v对只需要枚举u,v两个子树，计算大小为i的子树也只需要枚举一个子树以及一组u,v，这样复杂度就被优化为1到n所有可能形态总数的平方。这样能够搜索出的答案就可以扩大到$n=300$左右。

期望得分：80 分。

3.6 算法 6

有一种与算法 5 并列的优化，对于树的结构，考虑找到树的重心，那么每个子树的大小都不超过$\frac{n}{2}$了。先做$\frac{n}{2}$以内的算法 5，然后再用大小在$\frac{n}{2}$以内的子树u,v，组合算法 5 中的u,v子树对结构。最后答案一定是两个u,v子树对结构拼接起来。

通过暴力可以发现，n接近400时，可能的形态数增长速度已经相当慢了，所以算法 6 主要是常数上的一些优化，但是效果还是很好的，由此可以搜索出$n=400$以内的答案。

期望得分：100 分。

3.7 方案输出

由于最后输出的并不是图G，因此需要把图G变成无向图。

对于n较小的办法，可以使用题目中暗示的$O(n2^n)$做法，每次删掉最小的节点的终点标号最小的边。这个点入度一定为0，然后枚举最小的节点下一条出边在哪里。记录$dp_{i,S}$表示删掉了点集S，当前点i是最小节点的终点标号最小的边，是否存在方案。当然直接手算也可以。

而n较大时，选手应该已经发现答案是一棵树，而 w 状链也构成树结构，递归标号即可，复杂度$O(n^2)$或$O(n^3)$都可以接受。

四．参考资料

由于本题是在日常的 OI 学习中产生的问题，未参考其他资料。

五．命题总结

本题首先考察了选手对问题的转化能力，对图论问题的分析能力。

打表是本题涉及到的主要方法之一。本问题比较复杂，结论的证明较为困难。找出正确的解法主要靠选手的尝试，以及在遇到各种情况的应变能力。

最后本题考察了选手综合运用各种算法优化方式的能力。