回忆（memory）

【题目背景】

生活在题面里的他们，是一群怪异的少年。

对城市中修建道路需满足的基本物理限制熟视无睹，沉迷于十万个城市、百万条道路上的各种结构。

明明知道真正需要的数字庞大到无法计算，却偏要关心它模一个奇怪素数之后得到的结果。

如此智力超群的他们，却总是在自己提出的诡异的问题下败下阵来，把它们一股脑地丢给你们来做。

如今，他们长大了。他们学习到更普适的理论，习惯了更抽象的符号，不必再思考如此古怪的问题。但他们不曾料到，你们却以这些“无用”的问题为驱动，于计算机学科体系的一隅，开垦出了一片独属于 OI 的新天地。

有一天，他们各自回忆起了少年时期提出的问题。

【题目描述】

少年时，他们提出了 n 个问题，从 1 到 n 编号。一个问题总由一个更基础的问题衍生而来，因此问题之间构成了一个树形的结构：1 号问题是最基本的问题，也就是树的根节点，而其他问题都由其父亲节点对应的问题衍生而来。如果两个问题在树上相邻，则称这两个问题**彼此相关**。

少年时期的他们一共做了 m 次研究，第 i 次的研究从提出较为基本的问题 s_i 开始，将它不断地修改、推广，最终提出问题 t_i。这些研究满足 $s_i \neq t_i$ 且 s_i **必定是** t_i **的祖先**。即使研究的问题完全相同，从不同的角度研究会有不同的结果，因此**可能存在** $i \neq j, (s_i, t_i) = (s_j, t_j)$ **的情形**。

现在，他们正一轮轮地回忆着少年时提出的问题。在他们每一轮对问题的回忆中，他们首先回忆到 n 个问题中的任意一个。接下来，如果存在与当前回忆到的问题**彼此相关且在这轮回忆中没有被回忆到**的问题，那么他们可以将思绪从当前问题上切换到这些问题中的**任意一个**，并回忆到这个新的问题。他们可以不断地切换思绪，也可以在回忆到任何一个问题之后结束回忆。**每一轮回忆是独立的，也就是说一个问题可以被多轮回忆回忆起。**

如果在某一轮回忆中，他们**同时**回忆到了问题 s_i 和问题 t_i，则称第 i 次研究**被想起**。

为了更好地理解上述概念，考察以下例子：$n = 5$，问题 1 与问题 2,3 相关，问题 3 与问题 4,5 相关。一轮可能的回忆是：从问题 2 开始回忆，切换思绪到问题 1 ，再切换到问题 3，最终切换到问题 5 并结束回忆。如果 $m = 4$，$(s_1, t_1) = (1, 2), (s_2, t_2) = (1, 4), (s_3, t_3) = (s_4, t_4) = (3, 5)$，那么这轮回忆会让第 1 次、第 3 次和第 4 次研究被想起，而第 2 次研究不会被想起。

他们问你们的最后一个问题是：**如果每轮回忆的起点以及思绪的切换可以任意选择，最少需要多少轮回忆才能使所有的研究都被想起**。

【输入格式】

从文件 ***memory.in*** 中读入数据。

本题有多组测试数据。输入的第一行包含一个正整数 T，表示数据组数。

对于每组测试数据，第一行包含两个正整数 n, m，分别表示问题的个数和研究的个数。

接下来 $n-1$ 行每行包含两个正整数 u_i, v_i，表示问题 u_i 与问题 v_i 相关。

接下来 m 行每行包含两个正整数 s_i, t_i，描述一次研究。

【输出格式】

输出到文件 ***memory.out*** 中。

对于每组测试数据输出一行一个正整数，表示他们**最少**需要回忆的轮数使得 m 次研究均被想起。

【样例 1 输入】

```
2
5 5
1 2
3 1
3 4
5 3
1 2
1 4
3 5
3 5
3 4
10 5
1 2
3 1
3 4
5 3
6 5
7 8
5 7
```

```
7 9
9 10
1 2
3 5
5 6
7 8
9 10
```

【样例 1 输出】

```
2
2
```

【样例 1 解释】

样例中的第一组数据与题目描述所给的例子相同。一种可能的回忆方案为：

- 第一轮回忆中，从问题 2 开始，依次切换思绪到问题 $1, 3, 5$。此时第 $1, 3, 4$ 次研究被想起，但第 $2, 5$ 次没有。
- 第二轮回忆中，从问题 4 开始，依次切换思绪到问题 $3, 1$。此时第 $2, 5$ 次研究被想起。

第二组数据符合特性 A 的要求。一种可能的回忆方案为：第一次回忆依次回忆到 $2, 1, 3, 5, 6$，第二次回忆依次回忆到 $8, 7, 9, 10$。

【样例 2】

见选手目录下的 ***memory/memory2.in*** 与 ***memory/memory2.ans***。

这组数据满足了测试点 $1 \sim 4$ 的条件。

【样例 3】

见选手目录下的 ***memory/memory3.in*** 与 ***memory/memory3.ans***。

这组样例了特性 A 的条件。且除了后 3 组数据外，其余样例均满足 $n, m \leq 1000$。除了后 30 组数据外，其余样例均满足 $n, m \leq 30$。你也可以用这组样例完成对较小规模数据的测试。

【样例 4】

见选手目录下的 ***memory/memory4.in*** 与 ***memory/memory4.ans***。

这组样例满足了测试点 24 ~ 25 的条件。同第 3 组样例，本样例满足：除了后 3 组数据外，其余样例均满足 $n, m \leq 1000$。除了后 30 组数据外，其余样例均满足 $n, m \leq 30$。你也可以用这组样例完成对较小规模数据的测试。

【样例 5】

见选手目录下的 ***memory/memory5.in*** 与 ***memory/memory5.ans***。
这组样例满足了特性 B 的条件。

【样例 6】

见选手目录下的 ***memory/memory6.in*** 与 ***memory/memory6.ans***。
这组样例满足了特性 C 的条件。

【子任务】

本题共 25 个测试点。对于所有的测试点，$1 \leq n, m \leq 200000$，$1 \leq \sum n, \sum m \leq 500000$，$1 \leq u_i, v_i, s_i, t_i \leq n$，$s_i \neq t_i$。保证输入的 (u_i, v_i) 构成一棵树，s_i 在以 1 为根的树上是 t_i 的祖先。

<table>
<tr><th>测试点</th><th>规模限制</th><th>特性 A</th><th>特性 B</th><th>特性 C</th></tr>
<tr><td>1</td><td rowspan="4">$T \leq 3000$，$n \leq 50$，$m \leq 15$ 且最多有 5 组数据满足 $m \geq 10$</td><td rowspan="4">No</td><td rowspan="8">No</td><td rowspan="10">No</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td><td rowspan="5">$n, m \leq 100$，$\sum n^3, \sum m^3 \leq 3 \cdot 10^7$</td><td rowspan="2">Yes</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td><td rowspan="3">No</td><td>Yes</td></tr>
<tr><td>8</td><td rowspan="3">No</td><td>Yes</td></tr>
<tr><td>9</td><td rowspan="3">No</td></tr>
<tr><td>10</td><td rowspan="6">$n, m \leq 1000$，$\sum n^2, \sum m^2 \leq 3 \cdot 10^7$</td><td rowspan="2">Yes</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td><td rowspan="8">No</td><td>Yes</td></tr>
<tr><td>13</td><td rowspan="4">No</td><td>Yes</td></tr>
<tr><td>14</td><td rowspan="4">No</td></tr>
<tr><td>15</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td><td rowspan="9">$n, m \leq 2 \cdot 10^5$，$\sum n, \sum m \leq 5 \cdot 10^5$</td><td rowspan="2">Yes</td></tr>
<tr><td>18</td></tr>
<tr><td>19</td><td rowspan="5">No</td><td rowspan="2">Yes</td></tr>
<tr><td>20</td></tr>
<tr><td>21</td><td rowspan="3">Yes</td><td rowspan="5">No</td></tr>
<tr><td>22</td></tr>
<tr><td>23</td></tr>
<tr><td>24</td><td rowspan="2">No</td></tr>
<tr><td>25</td></tr>
</table>

特性 A：保证 n 为偶数，且树的结构为：对于任意正整数 $1 \leq i \leq \lfloor \frac{n}{2} \rfloor$，$2i$ 的父亲为 $2i-1$；若 $i \geq 2$，则 $2i-1$ 的父亲为 $2i-3$。

特性 B：保证对于所有的正整数 $1 \leq i \leq m$，s_i 为 t_i 的父亲。

特性 C：保证对于所有的正整数 $1 \leq i \leq m$，$s_i = 1$。

请注意，**测试点的难度与编号并没有直接关系**。

【评分方式】

对于每一组测试点，如果你的输出格式正确，且每一组数据输出的答案正确，那么你会获得 4 分。

否则，如果你的输出格式正确，且对于每一组数据，你输出的答案**与正确答案相等或者比正确答案大** 1，那么你将在此测试点上获得 3 分。

【提示】

请注意，为了取得部分分，你必须保证输出格式正确，即：输出恰好有 m 行，且每行是一个正整数。

此外，如果某组测试数据中你输出的结果比答案小 1 而不是大 1，那么你**不能**在该测试点获得 3 分。

本题部分测试点读入量较大。为了优化程序的总运行时间，我们建议你采用较为快速的读入方式。