

ACM/ICPC WORLD FINALS
2009 PROBLEM H
THE MINISTERS' MAJOR MESS

绍兴市第一中学 俞鼎力

2013 年 9 月 30 日

【问题描述】

在一个名为 Stanistan 的遥远国度，大臣们在做决策时遇到了很严重的问题。几周前的一次新的议案投票进程引发了这个问题，这个进程的每次会议中，有若干议案被投票。每个大臣会对某些议案投票表示赞成或反对。由于票数统计等一系列技术方案在设计上局限性，每个大臣只能对最多四个不相同的议案投票（但这几乎不成问题，因为大多数大臣只对少数议题感兴趣）。得到了这些投票之后，议案将被决定是否通过，使得每个大臣有大于一半的建议被满足。

许多聪明的读者也许已经发现，这个进程导致了各种问题。比方说，可能有多个方案符合要求，或者说没有一种方案符合要求，就算只有一种方案，又该怎样得到这个方案。

你的任务就是写一个程序来帮助大臣们解决这些问题。给出每个大臣的投票，你的程序需要指出是否每个大臣都能被满足，如果是，请指出哪些议案只能被通过或否决。

【输入格式】

从文件 *major.in* 中读入数据。

输入第一行两个数 n, m ，分别表示议案个数和大臣人数。

接下来 m 行给出大臣们的投票，每行第一个数 $k (1 \leq k \leq 4)$ ，表示这个大臣投的票数。接下来是 k 个投票，每个投票都是 $\langle \text{bill} \rangle \langle \text{vote} \rangle$ 的格式，其中 $\langle \text{bill} \rangle$ 是一个 1 到 n 的整数表示被投票的议案编号， $\langle \text{vote} \rangle$ 是 y 或 n，表示大臣的意见是通过还是否决。数据保证没有大臣会对同一个议案投票多次。

【输出格式】

输出到文件 *major.out* 中。

输出一行。如果不可能满足所有大臣，那么输出 impossible。否则

输出一个长度为 n 的字符串，如果编号为 i 的议案一定要通过，那么字符串的第 i 位为 y，如果一定不能通过，则为 n，否则为 ?。

【样例输入1】

```
5 2
4 2 y 5 n 3 n 4 n
4 4 y 3 y 5 n 2 y
```

【样例输出1】

```
?y??n
```

【样例输入2】

```
4 2
4 1 y 2 y 3 y 4 y
3 1 n 2 n 3 n
```

【样例输出2】

```
impossible
```

【数据规模与约定】

30%的数据 $n \leq 16$ 。

另 30%的数据 $m \leq 8$ 。

100%的数据 $1 \leq n \leq 100, 1 \leq m \leq 500$ 。