Pear-wise Voting

Problem ID: pearwise
CPU Time limit: 1 secor
Memory limit: 1024 ME
Difficulty: 4.3

Category: 4.2a, Finding
Hint: clever problem wc be reduced into basic gr traversal

Author: Deon Nicholas
Source: 2018 ICPC East-NA Regional Contest

Bob Roberts is ending his term as president of a local group of fruit aficionados call the Pear-wise Club. Being president is a plum position so many people are running, some of whom Bob thinks are just peachy, while others are just the pits. After some thought, Bob has amended the by-laws of the Pear-wise Club to use sequential pairwise voting to determine his successor.

Sequential pairwise voting works as follows: every voter presents a ballot which ranks the candidates from most to least favorable. Then two of the candidates are selected to hold a one-on-one contest with these ballots, using the relative positions of the candidates on each ballot to determine how each voter would vote in this contest. The winner of this contest then takes part in a second one-on-one contest with a third candidate; the winner of that takes part in a third contest with a fourth candidate, and so on. The winner of the election is the winner of the last one-on-one contest.

The order in which the candidates take part in the one-on-one contests is specified by the *agenda*, which is a pre-selected ordering of the candidates. The first two candidates on the agenda take part in the first contest; the winner of that contest then faces the third candidate on the agenda, an so on. For example, assume we had the following set of 13 ballots, each ranking five candidates A, B, C, D and E:

(4)	(3)	(3)	(2)	(1)
A	D	C	B	E
D	C	A	D	B
C	A	D	C	C
B	B	B	A	D
E	E	E	E	A

The number above each ballot indicates the number of voters who submitted that ballot. If the agenda is ABCDE, then sequential pairwise voting would proceed as follows: in the first contest between A and B, A wins $10-3$; A would then face C (the third candidate on the agenda) and here C wins $9-4$; C would then face D and D wins $9-4$; and finally D would face E and D would win $12-1$, so D would be the overall winner using this agenda. Using the reverse agenda – EDCBA – would result in A winning the election (we'll let you figure out the details of that one).

Now it wasn't out of any political science reasons that Bob picked sequential pairwise voting, or the coincidence of its name. He knows that the results are highly dependent on the agenda that's used and as current president, Bob gets to set the agenda! Since he's also aware of everyone's voting preferences, he's pretty sure that he can find an agenda so that his preference for who succeeds him will be elected. Just to be sure, he would like you to write a program which, when given a set of ballots as input, determines, for each candidate, whether there is an agenda which will result in his/her winning the election.

Input

Input begins with a line containing two positive integers n m ($n \le 26, m \le 2\,000$) indicating the number of candidates and the number of unique ballots, respectively. Candidates are labeled with the first n letters of the alphabet, uppercase. Each of the next m lines displays one of the m ballots. Each line starts with a positive integer p ($p \le 100$) indicating the number of voters who submitted the ballot, following by a string of n characters specifying the ballot. This upper-case string is a permutation of the first n characters of the English alphabet, with the first character in the string indicating the most favored candidate in the ballot, the second character the second-most favored, and so on. The total number of votes over all ballots will be odd, so there can't be a tie in any pairwise vote.

Output

For each candidate display the candidate's letter followed by a colon, followed by either the phrase `can win` if an agenda exists where the candidate can win, or the phrase `can't win` if no such agenda exists.

Help

Sample Input 1

```
5 5
4 ADCBE
3 DCABE
3 CADBE
2 BDCAE
1 EBCDA
```

Sample Output 1

```
A: can win
B: can't win
C: can win
D: can win
E: can't win
```

Sample Input 2

```
3 3
1 ABC
1 BCA
1 CAB
```

Sample Output 2

```
A: can win
B: can win
C: can win
```