D Disgruntled Diner

Time limit: 2s



Tickets allow Diana to oversee kitchen operations. Unsplash Licence by Daniel Bradley on Unsplash

Diana is head chef at the Batavian Authentic Prestigious Cuisine. All orders are logged to a central computer, but to organize the kitchen, each ordered item is also printed on a separate *ticket*. On the back of each ticket, Diana writes the corresponding table number, so servers can easily check where they need to deliver the food. When a ticket is completed, Diana pins it to a board. Of course, there could be duplicate tickets, since a table may order the same item multiple times.

Halfway through service, a disgruntled diner complains: "We have been waiting for hours, but so far we were *only* served tomato soup! Could you please hurry up?" Diana must address this immediately, since the reputation of the restaurant is at stake! However, in the past, some customers have been dishonest in order to demand expedited service. So, before Diana instructs the kitchen staff, the complaint must be verified.

Since the board is always up-to-date, verification is a matter of checking the relevant tickets. Let t and m be the table number and menu item that describe the complaint, respectively. Diana seeks to verify the following claim: "All pinned-up tickets for table t correspond to menu item m." In particular, if there are no completed tickets for table t, Diana considers the claim to be true – the customer may have misspoken, but they surely deserve extra attention!

Unfortunately, on the board, only one side of each ticket is visible (either the menu item or table number) and Diana does not have time to flip hundreds of tickets. However, by cross-referencing with the central computer, it may be possible to safely ignore certain tickets. Help Diana determine the minimum set of pinned-up tickets that need to be flipped. Or can you (dis)prove the claim without flipping any tickets? You must decide which tickets should be flipped before Diana starts doing so – she is too strained to make deductions on the fly.

Input

The input consists of:

- One line with two integers n and k ($1 \leq k \leq n \leq 500$), the number of orders in the computer and the number of pinned-up tickets.
- One line with n strings representing the orders in the computer, each string consisting of one English uppercase letter (the menu item, `A-Z`) and one digit (the table number, `0-9`).
- One line with k characters representing the pinned-up tickets, each character being either an English uppercase letter or a digit.
- One line with a digit t (`0-9`) and an English uppercase letter m (`A-Z`), specifying the claim: "All pinned-up tickets for table t correspond to menu item m."

The pinned-up tickets are guaranteed to correspond to (a subset of) the orders in the computer.

Output

If, without flipping any tickets, the claim is provably true or false, output "`true`" or "`false`", respectively. Otherwise, output the minimum number of pinned-up tickets that need to be flipped, followed by their 1-based indices, in any order.

Sample Input 1	**Sample Output 1**

`6 4`
`A1 A2 B1 B2 C1 C2`
`A B 1 2`
`1 A`
	`2`
	`3 2`

Sample Input 2	**Sample Output 2**

`5 4`
`A1 B1 C2 C1 A2`
`2 A B 1`
`1 A`
	`false`

Sample Input 3	**Sample Output 3**

`4 4`
`Z0 Z0 F9 F9`
`Z 0 9 9`
`4 F`
	`true`