

## Problem A. Aho

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

*All characters appearing in this and following statements are fictitious. Any resemblance to real persons, living or dead, is purely coincidental.*

This is an interactive problem.

Alfred and Margaret always play some weird string-guessing games when they spend a charming evening in the coffee shop “Donald’s place”.

This time, the rules are the following:

- The first player writes down a string  $S$  of a predefined length  $N$ . The first player also has a string  $T$  that is initially empty. Both strings contain only **lowercase English letters**.
- The second player can not see these strings during the whole game. However, the second player is allowed to ask whether the characters on any pair of positions in any of the strings are equal. For example, a question can look like the following: “Is the second character of string  $S$  equal to the fifth character of string  $T$ ?” Note that it is also allowed to compare two positions of the same string using this question.
- The game is played in  $M$  rounds. At the start of each round, the first player adds a single character to the end of string  $T$ .
- When a new character is added, the second player can ask **no more than five** questions. After that, the second player must say how many substrings of string  $T$  are equal to string  $S$ .

Margaret quickly noticed that Alfred always succeeds as the second player. She suspects there is a strategy that allows the second player to win regardless of what are the strings  $S$  and  $T$ . Could you figure it out?

### Input

When your program starts, it must read two integers  $N$  and  $M$  from the standard input ( $1 \leq N, M \leq 20\,000$ ).

Then  $M$  rounds of the game follow. In the  $i$ -th round, you can ask **up to five** questions in the form “<position1> <position2>”. Here, the description of each position is either in the form “s  $x$ ” where  $1 \leq x \leq N$  ( $x$ -th character of the string  $S$ ) or in the form “t  $y$ ” where  $1 \leq y \leq i$  ( $y$ -th character of the string  $T$ ). The response from the jury program is “Yes” if the characters on these positions are equal, and “No” otherwise.

To finish each round, you must output the answer in the form “\$  $k$ ” where  $k$  stands for the number of occurrences of string  $S$  in the current string  $T$ . After that, the string  $T$  will be automatically expanded by some unknown character, or the game finishes if it is the last round.

Do not forget to flush the standard output after each operation. Once you already reported all the  $m$  required numbers, you should exit the program, otherwise the result of the judgement is **undefined!**

## Examples

standard input	standard output
3 7	s 1 s 2
No	s 1 s 3
Yes	s 2 t 1
No	s 1 t 1
Yes	\$ 0
Yes	s 2 t 2
Yes	\$ 0
No	s 3 t 3
No	\$ 1
Yes	s 2 t 4
Yes	s 1 t 4
Yes	\$ 1
	s 1 t 5
	\$ 1
	s 2 t 6
	\$ 1
	s 3 t 7
	\$ 2

## Note

In the example above, the string  $S$  is initially “aba”, and the string  $T$  is expanded sequentially by characters “a”, “b”, “a”, “c”, “a”, “b”, “a”.