

## Problem G: Interactive String Guessing

- Time Limit: 2 sec

### Problem Statement

*This is an interactive problem.*

Your task is to write a program that guesses a secret string through repetitive queries and responses. The secret string consists of **a** and **b**, and its length is between **1** and **1,000**. Let **S** be the secret string. In a query, you specify a string **T**. In response to this, whether **T** is a substring of **S** or not will be returned. Here, **T** is said to be a substring of **S** when some contiguous part of **S** matches **T**. For example, **aba** is a substring of **babab** and **aba** but not of **abba**. You can make up to **1,100** queries.

### Interaction

You should start with sending a query to the standard output and receiving the response from the standard input. This interaction can be repeated a number of times. When you have become confident of your guess through these interactions, you can send your answer. A query should be in the following format, followed by a newline.

**? T**

Here, **T** is a non-empty string consisting of **a** and **b**, and its length is between **1** and **1,000**, inclusive. The response to the query is given from the standard input in the following format followed by a newline.

**R**

Here, **R** denotes the response to the query. **R** is **Yes** when **T** is a substring of the secret string **S**, and **R** is **No** when **T** is not a substring of the secret string **S**. However, if **T** does not satisfy the constraints, or the number of queries exceeds the limit, then **R** is **Invalid**. If the judge returns **Invalid**, your program is already considered incorrect, so terminate the program immediately.

You should send your answer to the standard output in the following format, followed by a newline.

**! T**

Here, **T** is the secret string you have identified, its length is between **1** and **1,000**, inclusive. You can send the answer only once, so you have to become sure of your guess through repeated interactions before sending the answer. After sending the answer, your program should terminate without any extra output.

### Notes on interactive judging

The verdict will be indeterminate if there is malformed output during the interaction or your program quits prematurely. Terminate the program immediately after printing the answer, or the verdict will be indeterminate. As some environments require flushing the output buffers, make sure that your outputs are actually sent. Otherwise, your outputs will never reach the judge.

The string **S** will be fixed before the first interaction and will not be changed according to your questions or other factors.

### Sample Interaction

Read	Write
	? aba
Yes	
	? baa
No	
	! abab