# BWT

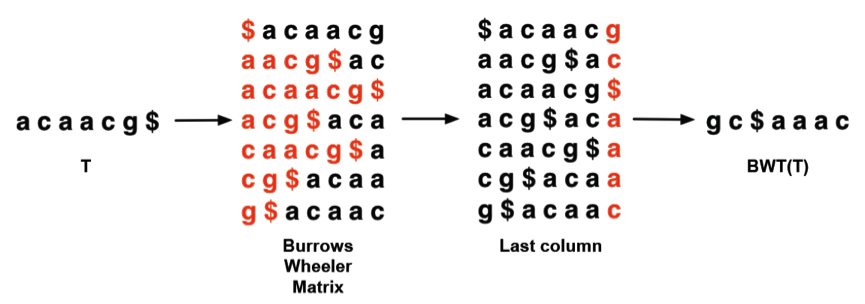
**Time limit** : 3 sec

**Memory limit** : 128 M

When the problem to match S string in T string is mentioned, people always put KMP, Aho-Corasick and Suffixarray forward. But Mr Liu tells Canoe that there is an algorithm called Burrows–Wheeler Transform(BWT) which is quite amazing and high-efficiency to solve the problem.

But how does BWT work to solve the matching S-in-T problem? Mr Liu tells Canoe the firstly three steps of it.

Firstly, we append the ‘$’ to the end of T and for convenience, we still call the new string T. And then for every suffix of T string which starts from i, we append the prefix of T string which ends at (i – 1) to its end. Secondly, we sort these new strings by the dictionary order. And we call the matrix formed by these sorted strings Burrows Wheeler Matrix. Thirdly, we pick characters of the last column to get a new string. And we call the string of the last column BWT(T). You can get more information from the example below.



Then Mr Liu tells Canoe that we only need to save the BWT(T) to solve the matching problem. But how and can it? Mr Liu smiles and says yes. We can find whether S strings like “aac” are substring of T string like “acaacg” or not only knowing the BWT(T)! What an amazing algorithm BWT is! But Canoe is puzzled by the tricky method of matching S strings in T string. Would you please help Canoe to find the method of it? Given BWT(T) and S string, can you help Canoe to figure out whether S string is a substring of string T or not?

**Input**

There are multiple test cases.

First Line: the BWT(T) string (1 <= length(BWT(T)) <= 100086).

Second Line: an integer n ( 1 <=n <= 10086) which is the number of S strings.

Then n lines comes.

There is a S string (n \* length(S) will less than 2000000, and all characters of S are lowercase ) in every line.

**Output**

For every S, if S string is substring of T string, then put out “YES” in a line. If S string is not a substring of T string, then put out “NO” in a line.

**Sample Input**

gc$aaac

2

aac

gc

**Sample Output**

YES

NO

**Hint**

A naïve method will not be accepted.