Problem 4 - ROP Chain (ROP)

In the last *Reply Cyber Security Challenge*, the most difficult exercises were those about binary security. So this year, we're asking the participants of the *Reply Code Challenge* to help the participants of the next *Reply Cyber Security Challenge*.

A key technique in the binary challenges is the *Return Oriented Programming* (**ROP**). This is where attackers can execute arbitrary code using only the codes available in the memory, in particular, by chaining (a *rop chain*) only suffixes of a limited set of strings (called *gadgets*).

More formally, given a string S and a set of N strings $G[0], \ldots, G[N-1]$ we're asking you which is the minimum number of gadgets (suffixes of the strings $G[i]$), you have to use to obtain the string S. Unfortunately, this operation is not always possible, and if there's no solution, the answer is -1.

Example:

Given the strings `G[2] = {missi, lippi}`

We can use these *gadgets*: `missi, issi, ssi, si, i, lippi, ippi, ppi, pi, i`

The string `S = "mississippi"` can be obtained using three gadgets: `missi` + `ssi` + `ppi`

The string `S = "sisi"` can be obtained using only two gadgets: `si` + `si`

Instead, the string `S = "lissi"` cannot be obtained in any way.

You can reuse the same gadget multiple times (each time, it will count +1 for the solution).

It is possible there are same gadgets from different strings, and you can use the ones you want without distinction.

Input data

The first line of the input file contains an integer **T**, the number of test cases to solve, followed by **T** testcases, numbered from **1** to **T**.

In each test case, the first line contains the integers **N**, the number of strings.

The second line contains the string **S**.

The following **N** lines contains the strings of **G**.

Output data

The output file must contain **T** lines. For each test case in the input file, the output file must contain a line with the words:

Case #t: g

where t is the test case number (from **1** to **T**) and g is the minimum number of gadgets to obtain the given string (or -1 if there is no solution).

Constraints

- $1 \leq \mathbf{T} \leq 20$.
- $1 \leq \mathbf{N} \leq 1\,000$.
- $1 \leq |\mathbf{S}| \leq 1\,000$.
- $1 \leq |\mathbf{G[i]}| \leq 1\,000$, for each $0 \leq i < N$.
- All the strings **S** and **G[i]** are formed by lower-case alphabetic characters.

Scoring

- **input 1** : $\mathbf{T} = 1$, $\mathbf{N} \leq 3$, $|\mathbf{S}| \leq 10$, $|\mathbf{G}| \leq 1$.
- **input 2** : $\mathbf{T} = 5$, $\mathbf{N} \leq 10$, $|\mathbf{S}| \leq 100$, $|\mathbf{G}| \leq 10$.
- **input 3** : $\mathbf{T} = 10$, $\mathbf{N} \leq 100$, $|\mathbf{S}| \leq 200$, $|\mathbf{G}| \leq 100$.
- **input 4** : $\mathbf{T} = 15$, $\mathbf{N} \leq 200$, $|\mathbf{S}| \leq 500$, $|\mathbf{G}| \leq 200$.
- **input 5** : $\mathbf{T} = 20$, $\mathbf{N} \leq 1\,000$, $|\mathbf{S}| \leq 1\,000$, $|\mathbf{G}| \leq 1\,000$.

Examples

input	output
3 2 mississippi missi lippi 2 sisi missi lippi 2 lissi missi lippi	Case #1: 3 Case #2: 2 Case #3: -1