C – Match Maker

In Computer Science, *pattern matching* is the act of checking if a certain sequence conforms (matches) a given pattern. Patterns are usually specified using a language based on *regular expression*. In this problem, we'll use a simple regular expression to express patterns on sequences of decimal digits. A pattern is a sequence of one or more decimal digits '0'. . . '9', asterisks '*', and hash signs '#'. A '*' denotes a sequence of an even number of digits, whereas a '#' denotes a sequence of an odd number of digits. For example, the pattern "129" only matches the sequence 129. The pattern "1*3" matches all sequences beginning with 1, ending with 3, and having an even number of decimal digits between the first and last digits. As another example, the pattern "#55" matches the sequences 155, 12355, 1234555, but none of the sequences 55, 1255, 123455.

Your task is to write a program to find if a given sequence matches a given pattern

Input Format

Your program will be tested on one or more data sets. Each data set contains a single pattern and one or more sequences to match. The first line of each data set specifies the pattern, and the remaining lines specify the sequences to match against that pattern. The end of a data set (except the last) is identified by the word "END" (without the double quotes.) The end of the last data set is identified by the word "QUIT". All lines are 100,000 characters long or shorter.

Output Format

```
k.s._result
```

Where k is the test case number (starting at one,) and s is the sequence number (starting at one within each test case,) and result is either the word "match" if the given string matches the pattern, or the word "not" if it doesn't.

Sample Input	Sample Output
129 1299 129 1129 END 1*3 123 1223 END #55 155 12355 55 1255 QUIT	1.1. not 1.2. match 1.3. not 2.1. not 2.2. match 3.1. match 3.2. match 3.3. not 3.4. not