Problem 7: Jumble!

 "Jumble"® is a word puzzle that appears in many daily newspapers. Each word in the puzzle has its letters rearranged to make it look like it might be another word, but isn't. The challenge in the puzzle is to unscramble the rearranged letters to uncover the original word.

In this problem you are to write a utility program that evaluates potentially scrambled words to determine their "degree of difficulty."

One feature that makes a particular reordering of the letters look "hard" is if the reordered letters look like a "real" word. For example, consider reordering the letters of the word "IRONY." Two possible reorderings are "ROYIN" and "NRYIO." The first of these is a "harder" reordering of the letters, since it confuses the brain into thinking that it is already a word (that is, it is more pronounceable).

Another aspect that makes a scrambling "harder" is when the letters in the scrambled word do not appear in the same position as in the original word. For example, the scrambled word "RIONY" is not a very good scrambling of "IRONY", even though it is pronounceable.

Your task is to write a program that scores proposed scrambles as "good," "fair," "poor," or "not." Your program will read pairs of words, the first of which is the original word, and the second of which is the proposed scrambled word. A scrambled word is "good" if none of its letters are in their original position and it appears "real" (more details are provided later). A scramble is "poor" if it doesn't look "real" and either the first letter is in the original location, or any two other letters are in their original locations. If the proposed scrambled word isn't scrambled at all, then it is scored as "not." All other scrambles are scored as "fair."

How do we know if a word looks "real?" We use an extremely crude, but easily usable heuristic that requires that the scrambled word must alternate between vowels (including 'Y' for these purposes) and consonants. An exception is made in certain cases, however. Any of the following sequences of letters are allowed to appear without making the word appear "unreal."

```
AI AY EA EE EO IO OA OO OY YA YO YU

BL BR CH CK CL CR DR FL FR GH GL GR

KL KR KW PF PL PR SC SCH SCR SH SHR

SK SL SM SN SP SQ ST SW TH THR TR

TW WH WR
```

Also, all double consonants are allowed. No other combinations are allowed, so "SWR" would not be good, even though both "SW" and "WR" are both acceptable.

Input

The input will contain multiple cases. The words for each case will appear on two consecutive lines, starting in the first position and ending with the first non-alphabetic character (which may be the end of line). The second word of each pair will always be an anagram of the first word, meaning it has the same letters (but possibly in a different order). The end of input is indicated by the end of file.

Output

For each pair of words, display a single line that indicates how well the second word scores as a scramble of the first. Use a format very similar to that shown in the example below. Separate the output for each pair of words with a blank line.

Sample Input	Expected Output
`SPAM`	`"MAPS" is a fair scramble of "SPAM"`
`MAPS`	
`IRONY`	`"RIONY" is a fair scramble of "IRONY"`
`RIONY`	
`IRONY`	`"ONYRI" is a good scramble of "IRONY"`
`ONYRI`	
`IRONY`	`"IRONY" is not a scramble of "IRONY"`
`IRONY`	
`IRONY`	`"INOYR" is a fair scramble of "IRONY"`
`INOYR`	
`IRONY`	`"IOYRN" is a poor scramble of "IRONY"`
`IOYRN`	
End of file	