

11339 Sonnet Rhyme Verifier

According to Wikipedia

The term *sonnet* derives from the Provençal word *sonet* and the Italian word *sonetto*, both meaning “little song”. By the thirteenth century, it had come to signify a poem of fourteen lines that follows a strict rhyme scheme and logical structure. The conventions associated with the sonnet have evolved over its history. The writers of sonnets are known as *sonneteers*.

Written in Spanish, sonnets have a well defined structure over the rhymes. Moreover, these days sonneteers are scarce as good weather and the rules governing the rhymes are rather permissive. Let us consider the following poem:

```
ES ELLA
Locura, intensa y sin medida
andando errante te avisté,
una nota en tus manos encontré:
rumor de un amor y su partida.
Adelante, te diré de su vida
aunque en mi regazo ya no esté,
fueron sus labios con que tropecé
los que te llamaron despavorida.
Ojos negros, pestañas en un ramo,
rizos finos, humor, siempre bella;
!`es ella, amiga, a quien amo!
Zagales de secretos sus estrellas,
cómplices la lluvia y el álamo;
... es ella, nuestra musa, !`es ella!
```

This poem must be consider as sonnet because it meets the following rules:

1. It has 14 lines.
2. Its rhymes have the structure $ABBAABBACDCDCD$, where A , B , C and D correspond to the suffixes *ida*, *é*, *amo*, and *ella* of the given line. In general, we are to allow any rhyme structure as long as it follows one of the patterns $ABBAABBACDECDE$, $ABBAABBACDEDCE$ or $ABBAABBACDCDCD$.
3. It has hendecasyllable meter.

Today you are to help the local Spanish School of Sonneteers (SSS) writing a program that verifies if a given composition sticks to the first two rules of a Spanish sonnet. The last rule is to be checked by the SSS’s master sonneteer once the given poem passes the tests exercised by the verifier you are about to code.

For the purpose of your work, two lines rhyme if, after *cleaning* them, they have the same suffix. Given a line in the sonnet, its clean version is the result of deleting blanks, punctuation symbols (*¡*, *!*, *,*, *.*, *:*, *;*, *¿*, *?*, *-*) and the last *s* (if there is one) from the end of the original line.

Input

The input consists of several instances of the problem, each one occurring separated by a new line. Each instance consists of $n > 2$ lines: the first one consists of a list of s suffixes ($4 \leq s \leq 5$), separated by a blank character. The suffixes are to be called A , B , C , D and E according to the order in which they occur and the total number of them. The second line contains the title of the composition. The following $n - 2$ lines contain the actual lines of the composition. Each line in the sonnet has at most 80 characters. You can safely assume that, in a case, a given suffix is not suffix of another one.

Output

For each problem instance, and according to the order of the input, your program should print a sentence of the form

`< NAME >: < STRUCTURE >`

if the given composition adheres to the first two rules of a spanish sonnet or

`< NAME >: Not a chance!`

in other case, where `< NAME >` is to be replaced by the name of the composition and `< STRUCTURE >` corresponds to the structure of the rhyme, and is given by the order in which the suffixes occur.

Sample Input

```
ida é amo ella
ES ELLA
Locura, intensa y sin medida
andando errante te avisté,
una nota en tus manos encontré:
rumor de un amor y su partida.
Adelante, te diré de su vida
aunque en mi regazo ya no esté,
fueron sus labios con que tropecé
los que te llamaron despavorida.
Ojos negros, pestañas en un ramo,
rizos finos, humor, siempre bella;
!`es ella, amiga, a quien amo!
Zagales de secretos sus estrellas,
cómplices la lluvia y el álamo;
... es ella, nuestra musa, !`es ella!
```

```
ura ima illa eto
OTr0
Esta composición posiblemente rima
pero nunca será un soneto
con esta frase ya no sirve
ni tampoco porque está incompleto.
```

Sample Output

```
ES ELLA: ABBAABBACDCDCD
OTr0: Not a chance!
```