

ACM Pacific NW Region Programming Contest 13 November 2004

Problem B Zipper

Problem description

Given three strings, you are to determine whether the third string can be formed by combining the characters in the first two strings. The first two strings can be mixed arbitrarily, but each must stay in its original order.

For example, consider forming "tcraete" from "cat" and "tree":

```
String A: cat
String B: tree
String C: tcraete
```

As you can see, we can form the third string by alternating characters from the two strings. As a second example, consider forming "catrtee" from "cat" and "tree":

```
String A: cat
String B: tree
String C: catrtee
```

Finally, notice that it is impossible to form "cttaree" from "cat" and "tree".

The Input (from file b.in)

The first line of input file, b.in, contains a single positive integer from 1 through 1000. It represents the number of data sets to follow. The processing for each data set is identical. The data sets appear on the following lines, one data set per line.

For each data set, the line of input consists of three strings, separated by a single space. All strings are composed of upper and lower case letters only. The length of the third string is always the sum of the lengths of the first two strings. The first two strings will have lengths between 1 and 200 characters, inclusive.

The Output (to stdout)

For each data set, print:

```
Data set n: yes
```

if the third string can be formed from the first two, or

```
Data set n: no
```

if it cannot. Of course n should be replaced by the data set number. See the sample output below for an example.

**ACM Pacific NW Region Programming Contest
13 November 2004**

Sample input

```
3
cat tree tcraete
cat tree catrtee
cat tree cttaree
```

Sample output

```
Data set 1: yes
Data set 2: yes
Data set 3: no
```