Proper Subsets (prob9)

A set is a collection of elements, or members. Order and count do not matter, so the set {1, 2, 3} is the same as the sets {3, 2, 1} and {1, 2, 1, 3, 3, 1, 2}.

If A and B are sets and every element of A is also an element of B, then A is a subset of (or is included in) B, denoted by $A \subseteq B$. For example $\{1, 2, 3\} \subseteq \{4, 3, 2, 1\}$ and $\{1, 2, 3\} \subseteq \{3, 2, 1\}$.

If A is a subset of B, but A is not equal to B (i.e. there exists least one element of B not contained in A), then A is also a proper subset of B, denoted by $A \subset B$. For example
$\{1, 2, 3\} \subset \{4, 3, 2, 1\}$ but $\{1, 2, 3\} \not\subset \{3, 2, 1\}$.

Write a program to determine whether one set of identifiers is a proper subset of another.

Input

The input will begin with a positive integer that represents the number of test cases. Each test case will consist of two sets, one per line. The sets will begin with a curly brace and have 0 to 20 identifiers in the set. Identifiers will be 1 to 15 characters. The characters in the identifiers will be just letters and digits. There will be one comma between identifiers and a closed curly brace at the end of the list of all identifiers. There may be blanks between the braces and the identifier and between the identifiers and the commas. Case is not significant in the identifiers, so "BEAR" and "bear" should be considered to be the same identifier. There may be duplicates in the lists.

Output

The output for each case will be the case number, followed by a colon, a space and the word YES indicating if the second set is a proper subset of the first set and NO otherwise.

Sample input

```
4
{Apple,Grape,Orange,Banana,Pear}
{ORANGE,GRAPE}
{Apple, Grape, Pear, Orange, GrapeFruit, Apple}
{ORANGE, BANANA, GRAPE, Grape, Grape, Grape, grape}
{ Grapefruit }
{Grape}
{Door,Shelf,Knob,Window}
{Door,Shelf,Knob,Window}
```

Output (corresponding to sample input)

```
1: YES
2: NO
3: NO
4: NO
```