Abbreviation

You can perform the following operations on the string, a:

1. Capitalize zero or more of a's lowercase letters.
2. Delete all of the remaining lowercase letters in a.

Given two strings, a and b, determine if it's possible to make a equal to b as described. If so, print YES on a new line. Otherwise, print NO.

For example, given $a = \mathrm{AbcDE}$ and $b = \mathrm{ABDE}$, in a we can convert b and delete c to match b. If $a = \mathrm{AbcDE}$ and $b = \mathrm{AFDE}$, matching is not possible because letters may only be capitalized or discarded, not changed.

Function Description

Complete the function $abbreviation$ in the editor below. It must return either YES or NO.

abbreviation has the following parameter(s):

- *a*: the string to modify
- *b*: the string to match

Input Format

The first line contains a single integer q, the number of queries.

Each of the next q pairs of lines is as follows:
- The first line of each query contains a single string, a.
- The second line of each query contains a single string, b.

Constraints

- $1 \leq q \leq 10$
- $1 \leq |a|, |b| \leq 1000$
- String a consists only of uppercase and lowercase English letters, ascii[A-Za-z].
- String b consists only of uppercase English letters, ascii[A-Z].

Output Format

For each query, print YES on a new line if it's possible to make string a equal to string b. Otherwise, print NO.

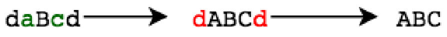
Sample Input

```
1
daBcd
ABC
```

Sample Output

```
YES
```

Explanation



We have $a =$ `daBcd` and $b =$ `ABC`. We perform the following operation:

1. Capitalize the letters `a` and `c` in a so that $a =$ `dABCd`.
2. Delete all the remaining lowercase letters in a so that $a =$ `ABC`.

Because we were able to successfully convert a to b, we print `YES` on a new line.