# String it out

Let **A** and **B** be two strings made up of alphabets such that $\mathbf{A = A_{[1-n]}}$, $\mathbf{B = B_{[1-m]}}$. We say **B** is a subsequence of **A** if there exists a sequence of indices $\mathbf{i_1 < i_2 <..m}$ of **A** such that $\mathbf{A[i_k] = B[k]}$.

Given **B[1-m]**, a string of characters from some alphabets, **B^i** is defined as string with the characters of **B** each repeating **i** times. For example, **(abbacc)^3 = aaabbbbbbaaaccccccc**. Also, **B^0** is the empty string.

Given strings **X**, **Y** made up of characters from **'a' - 'z'** find the maximum value of **M** such that **X^M** is a subsequence of **Y**.

## Input

- The first line of the input contains a positive integer **t <= 20**, denoting the no. of test cases.
- The following **2t** lines contain the value of **X** and **Y** for the cases.
- The description of the test cases follow one after the other.
    - Line **2k** contains the value of **X** for case **k**; **(1 <= k <= t)**
    - Line **2k+1** contains the value of **Y** for case **k**; **(1 <= k <= t)**.
    - The no. of characters in **X** , **Y** will be **<= 500010**.

## Output

The output must contain **t** lines, each line corresponding to a test case. The value on the $\mathbf{k^{th}}$ line should be the value of **M** for the $\mathbf{k^{th}}$ pair of **X** and **Y**.

## Example

**Input:**
3
abc
aabbcc
abc
bbccc
abcdef
abc

**Output:**
2
0
0