String Change

Problem Statement

You are given two string s and t, and a permutation A of $|s|$ integers, i.e. an array consisting of the integers in $[1, |s|]$ in a certain order.

You are supposed to transform the string s into the string t after deleting characters at certain positions of s (it is guaranteed that you can do this after deleting some characters from s).

But your friend, Mridul loves to delete characters as well! The problem is, he will delete the characters exactly in the order of the permutation A.

Since you do not want to make Mridul sad, find the maximum number of letters that he can delete, so that you can stop him (if required) at some point and finish deleting the rest of the letters in such a way, that the resulting string would be t.

NOTE: After deleting a character, the numbering of the positions does **not** change.

Input

The first line of input is the string s.
The second line of input is the string t.
The third line of input contains $|s|$ integers, which is the permutation A.

Output

Print the maximum number of deletions that Mridul can perform.

Sample input

> ababcba
> abb
> 5 3 4 1 7 6 2

Sample output

> 3

Explanation

After deleting the characters at position $5, 3$, and 4, the string would result in:

> abba

After this, you can delete the character at position 7 yourself to end up with the string:

> abb

If you allow Mridul to perform any more than 3 operations, it can be proved that achieving the string abb would be impossible, hence the output is 3.