You have a string of lowercase English alphabetic letters. You can perform two types of operations on the string:

1. *Append* a lowercase English alphabetic letter to the end of the string.
2. *Delete* the last character in the string. Performing this operation on an empty string results in an empty string.

Given an integer, k, and two strings, s and t, determine whether or not you can convert s to t by performing *exactly* k of the above operations on s. If it's possible, print Yes. Otherwise, print No.

For example, strings $s = [a, b, c]$ and $t = [d, e, f]$. Our number of moves, $k = 6$. To convert s to t, we first delete all of the characters in 3 moves. Next we add each of the characters of t in order. On the 6th move, you will have the matching string. If there had been more moves available, they could have been eliminated by performing multiple deletions on an empty string. If there were fewer than 6 moves, we would not have succeeded in creating the new string.

Function Description

Complete the *appendAndDelete* function in the editor below. It should return a string, either Yes or No.

appendAndDelete has the following parameter(s):

- *s*: the initial string
- *t*: the desired string
- *k*: an integer that represents the number of operations

Input Format

The first line contains a string s, the initial string.
The second line contains a string t, the desired final string.
The third line contains an integer k, the number of operations.

Constraints

- $1 \leq |s| \leq 100$
- $1 \leq |t| \leq 100$
- $1 \leq k \leq 100$
- s and t consist of lowercase English alphabetic letters, $\text{ascii}[a\text{-}z]$.

Output Format

Print Yes if you can obtain string t by performing exactly k operations on s. Otherwise, print No.

Sample Input 0

```
hackerhappy
hackerrank
9
```

Sample Output 0

```
Yes
```

Explanation 0

We perform 5 delete operations to reduce string s to hacker. Next, we perform 4 append operations (i.e., r, a, n, and k), to get hackerrank. Because we were able to convert s to t by performing exactly $k = 9$ operations, we print Yes.

Sample Input 1

```
aba
aba
7
```

Sample Output 1

```
Yes
```

Explanation 1

We perform 4 delete operations to reduce string s to the empty string (recall that, though the string will be empty after 3 deletions, we can still perform a delete operation on an empty string to get the empty string). Next, we perform 3 append operations (i.e., `a`, `b`, and `a`). Because we were able to convert s to t by performing exactly $k = 7$ operations, we print `Yes`.

Sample Input 2

```
ashley
ash
2
```

Sample Output 2

```
No
```

Explanation 2

To convert `ashley` to `ash` a minimum of 3 steps are needed. Hence we print `No` as answer.