LCS Returns

Given two strings, a and b, find and print the total number of ways to insert a character at any position in string a such that the length of the Longest Common Subsequence of characters in the two strings increases by one.

Input Format

The first line contains a single string denoting a.
The second line contains a single string denoting b.

Constraints

Scoring

- $1 \leq |a|, |b| \leq 5000$
- Strings a and b are alphanumeric (i.e., consisting of arabic digits and/or upper and lower case English letters).
- The new character being inserted must also be alphanumeric (i.e., a digit or upper/lower case English letter).

Subtask

- $1 \leq |a|, |b| \leq 1000$ for 66.67% of the maximum score.

Output Format

Print a single integer denoting the total number of ways to insert a character into string a in such a way that the length of the longest common subsequence of a and b increases by one.

Sample Input

```
aa
baaa
```

Sample Output

```
4
```

Explanation

The longest common subsequence shared by $a =$ **"aa"** and $b =$ **"baaa"** is `aa`, which has a length of 2. There are two ways that the length of the longest common subsequence can be increased to 3 by adding a single character to a:

1. There are 3 different positions in string a where we could insert an additional `a` to create longest common subsequence `aaa` (i.e., at the beginning, middle, and end of the string).
2. We can insert a `b` at the beginning of the string for a new longest common subsequence of `baa`.

As we have $3 + 1 = 4$ ways to insert an alphanumeric character into a and increase the length of the longest common subsequence by one, we print 4 on a new line.