NOIP '15 P5 - Substring

Time limit: 2.0s   **Memory limit:** 512M

You have two non-empty strings A and B made out of lowercase letters. You can build new strings as follows: take k non-overlapping non-empty substrings of A and arrange them in a new string in the order they were initially in A. Find how many of the strings that can be built in such a way are equal to B.

Constraints

Test Case	n	m	k
1	$n \leq 500$	$m \leq 50$	$k = 1$
2			$k = 2$
3			
4			$k = m$
5			
6			$k \leq m$
7			
8	$n \leq 1\,000$	$m \leq 100$	
9			
10		$m \leq 200$	
11			
12			

Input Specification

The first line will contain n, m, and k, the length of string A, the length of string B, and the number of substrings to take from A, respectively.

The second line will contain a string of length n, string A.

The third line will contain a string of length m, string B.

Output Specification

Output the number of ways to select k non-overlapping non-empty substrings of A such that their concatenation in order equals B.

Since the answer can be very large, output the answer modulo $10^9 + 7$.

Sample Input 1

```
6 3 1
aabaab
aab
```

Sample Output 1

```
2
```

Sample Input 2

```
6 3 2
aabaab
aab
```

Sample Output 2

```
7
```

Sample Input 3

```
6 3 3
aabaab
aab
```

Sample Output 3

7