Juiced String

Description

Jeffrey has a string X. He'll perform a few operations to turn the string X into another string Y.

There are only one operation he can use: insert one character into the current string.

For example, if X is `ace` and Y is `abcde`. Jeffrey owns `ace` at beginning, then he can insert a character `b` to get the string `abce`; after that, he can insert a character `d` to `abce` and get the string `abcde`, which matches the string Y.

If we concatenates all the strings Jeffrey owns after each operation in the order, we can get a longer string called footprint. In the above example, the footprint of Jeffrey's operations is `abceabcde` (gained by concatenating `abce` and `abcde`).

Here comes your task: given the strings X and Y, please compute how many different footprints are possible when Jeffrey turns the string X to string Y.

The answer may be large, so please print its remainder modulo $10^9 + 7$.

Input

The first line contains an integer T indicating the total number of test cases.

For each test case, there are two lines. The first line contains the string X and the second line contains the string Y.

- $1 \le T \le 70$
- $1 \le |X| < |Y| \le 50$
- Strings X and Y contain only lowercase English letters.
- It is possible for Jeffrey to turn the string X to string Y only with the insert operation.

Output

For each test case, output one integer indicating the number of different footprints modulo $10^9 + 7$.

Sample Input

```
2
ace
abcde
gg
ggggg
```

Sample Output

```
2
1
```