

# Count the strings

A letter is a lower case english alphabet (i.e it can be any symbol from 'a' to 'z').

Letters made of cheese are available in the market these days. Rat Ronnie bought many letters from the market. After placing these letters one after another in a straight line (i.e a string of letters) Rat Ronnie went to sleep.

Rat Rocky who is always hungry, ate some of the cheese letters and escaped. Now Rat Ronnie is very furious. The string of letters that she had created was to be used in one of her science experiments. She doesnt remember the original string anymore. She wants you to find out the number of possible strings that she could have formed before Rat Rocky ate some of the characters.

(See examples and Notes for more clarification)

## Input

First line contains T , number of test cases. Next T lines will contain a string of characters which may have any number of spaces anywhere in the string (including the beginning). The last character of the input string will not be a space. Each input string terminates with a newline character.

A space in the input string denotes an unknown character (i.e a character that Rat Rocky ate)

$T \leq 100$

Each string will not exceed 100 characters.

## Output

For each test case output the number of strings that Rat Ronnie could have formed on a new line. As the answer may be huge, output the answer modulo 10000007.

## Notes

a modulo m : means the remainder after a is divided by m. It is also denoted by  $a \% m$ .

If a,b are non negative integers, then the following hold

- $(a+b)\%m = ((a\%m) + (b\%m))\%m$
- $(a*b)\%m = ((a\%m) * (b\%m))\%m$

## Example

### Input:

```
4
abc
deg
a c
z d
```

### Output:

```
1
```

26  
26  
676

**Explanation**

For the first test case, as there are no spaces in the string, it is the only possible string Rat Ronnie could have written.  
For the third test case "a c" the middle character can be any of the lower case 26 english alphabets.