

Shashank loves strings, but he loves palindromic strings the most. He has a list of  $n$  strings,  $A = [a_0, a_1, \dots, a_{n-1}]$ , where each string,  $a_i$ , consists of lowercase English alphabetic letters. Shashank wants to count the number of ways of choosing non-empty [subsequences](#)  $s_0, s_1, s_2, \dots, s_{n-1}$  such that the following conditions are satisfied:

1.  $s_0$  is a subsequence of string  $a_0$ ,  $s_1$  is a subsequence of string  $a_1$ ,  $s_2$  is a subsequence of string  $a_2$ ,  $\dots$ , and  $s_{n-1}$  is a subsequence of string  $a_{n-1}$ .
2.  $s_0 + s_1 + s_2 + \dots + s_{n-1}$  is a palindromic string, where  $+$  denotes the string concatenation operator.

You are given  $q$  queries where each query consists of some list,  $A$ . For each query, find and print the number of ways Shashank can choose  $n$  non-empty subsequences satisfying the criteria above, modulo  $10^9 + 7$ , on a new line.

**Note:** Two subsequences consisting of the same characters are considered to be different if their characters came from different indices in the original string.

### Input Format

The first line contains a single integer,  $q$ , denoting the number of queries. The subsequent lines describe each query in the following format:

- The first line contains an integer,  $n$ , denoting the size of the list.
- Each line  $i$  of the  $n$  subsequent lines contains a non-empty string describing  $a_i$ .

### Constraints

- $1 \leq q \leq 50$
- $1 \leq n \leq 50$
- $\sum_{i=0}^{n-1} |a_i| \leq 1000$  over a test case.

For 40% of the maximum score:

- $1 \leq n \leq 5$
- $\sum_{i=0}^{n-1} |a_i| \leq 250$  over a test case.

### Output Format

For each query, print the number of ways of choosing non-empty subsequences, modulo  $10^9 + 7$ , on a new line.

### Sample Input 0

```
3
3
aa
b
aa
3
a
b
c
2
abc
abc
```

### Sample Output 0

```
5
0
9
```

### Explanation 0

The first two queries are explained below:

1. We can choose the following five subsequences:

1.  $s_0 = \text{"a"}, s_1 = \text{"b"}, s_2 = \text{"a"}$ , where  $s_0$  is the first character of  $a_0$  and  $s_2$  is the first character of  $a_2$ .
2.  $s_0 = \text{"a"}, s_1 = \text{"b"}, s_2 = \text{"a"}$ , where  $s_0$  is the second character of  $a_0$  and  $s_2$  is the second character of  $a_2$ .
3.  $s_0 = \text{"a"}, s_1 = \text{"b"}, s_2 = \text{"a"}$ , where  $s_0$  is the first character of  $a_0$  and  $s_2$  is the second character of  $a_2$ .
4.  $s_0 = \text{"a"}, s_1 = \text{"b"}, s_2 = \text{"a"}$ , where  $s_0$  is the second character of  $a_0$  and  $s_2$  is the first character of  $a_2$ .
5.  $s_0 = \text{"aa"}, s_1 = \text{"b"}, s_2 = \text{"aa"}$

Thus, we print the result of  $5 \bmod (10^9 + 7) = 5$  on a new line.

2. There is no way to choose non-empty subsequences such that their concatenation results in a palindrome, as each string contains unique characters. Thus, we print **0** on a new line.