

# N-Palindromes

Time limit: 2000 ms

Memory limit: 512 MB

Alice thinks that contest problem authors' obsession with palindromes is misplaced. She is much fonder of  $n$ -palindromes, which are words that are palindromes when the characters at exactly  $n$  positions are changed.

For example, Alice knows that her name (in lowercase) is a 2-palindrome, because she can create any of the following palindromes from her name by changing 2 characters: `alila`, `acica`, `elile`, `ecice`.

She also knows that her name is a 3-palindrome, because she can create palindromes by changing characters at 3 positions, e.g. `ecace` and `zliilz`. However, this is only a partial list, and she wants your help in determining the total number of such palindromes.

## Standard input

The input starts with an integer  $t$ , on a line by itself, which gives the number of test cases.

Each test case is made up of an integer  $n$  followed by a lowercase string.

## Standard output

For each test case, you should output, on a line by itself, the total number of palindromes that can be created by changing exactly  $n$  characters of the given string. Since this number may be very large, you should output the number modulo  $10^9 + 7$ .

## Constraints and notes

- $1 \leq t \leq 20$
- $1 \leq n \leq \text{length of string} \leq 500$
- The characters of an  $n$ -palindrome, including the  $n$  replacement characters, must all be lowercase English letters.

### Input

```
3
2 alice
1 racecar
3 alice
```

### Output

```
4
25
196
```

### Explanation

The problem statement lists the four palindromes that can be made from the string `alice`, by changing 2 characters.

Since you can only change one character in `racecar`, you are constrained to changing the middle letter. This character can be changed to any of the 25 letters other than `e`.

For the last testcase, Alice has found that there are 196 palindromes that can be made from her name, by changing 3 characters.