C. Largest Subsequence

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Given is a string s of length n. In one operation you can select the lexicographically largest[†] subsequence of string s and cyclic shift it to the right[‡].

Your task is to calculate the minimum number of operations it would take for s to become sorted, or report that it never reaches a sorted state.

[†] A string a is lexicographically smaller than a string b if and only if one of the following holds:

- a is a prefix of b, but $a \neq b$;
- In the first position where a and b differ, the string a has a letter that appears earlier in the alphabet than the corresponding letter in b.

[‡] By cyclic shifting the string $t_1 t_2 \ldots t_m$ to the right, we get the string $t_m t_1 \ldots t_{m-1}$.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \leq n \leq 2 \cdot 10^5$) — the length of the string s.

The second line of each test case contains a single string s of length n, consisting of lowercase English letters.

It is guaranteed that sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output a single integer — the minimum number of operations required to make s sorted, or -1 if it's impossible.

Example

input Copy

```
6
5
aaabc
3
acb
3
bac
4
zbca
15
czddeneeeemigec
13
cdefmopqsvxzz
```

output Copy

```
0
1
-1
2
6
0
```

Note

In the first test case, the string s is already sorted, so we need no operations.

In the second test case, doing one operation, we will select `cb` and cyclic shift it. The string s is now `abc` which is sorted.

In the third test case, s cannot be sorted.

In the fourth test case we will perform the following operations:

- The lexicographically largest subsequence is `zca`. Then s becomes `abzc`.
- The lexicographically largest subsequence is `zc`. Then s becomes `abcz`. The string becomes sorted.

Thus, we need 2 operations.