

1. Palindrome Index

Given a string of lowercase letters in the range `ascii[a-z]`, determine the index of a character that can be removed to make the string a [palindrome](#). There may be more than one solution, but any will do. If the word is already a palindrome or there is no solution, return `-1`. Otherwise, return the index of a character to remove.

Example

`s = "bcbc"`

Either remove 'b' at index 0 or 'c' at index 3.

Function Description

Complete the `palindromeIndex` function in the editor below.

`palindromeIndex` has the following parameter(s):

- *string s*: a string to analyze

Returns

- *int*: the index of the character to remove or `-1`

Input Format

The first line contains an integer *q*, the number of queries.

Each of the next *q* lines contains a query string *s*.

Constraints

- $1 \leq q \leq 20$
- $1 \leq \text{length of } s \leq 10^5 + 5$
- All characters are in the range `ascii[a-z]`.

Sample Input

STDIN	Function
-----	-----
3	q = 3
aaab	s = 'aaab' (first query)
baa	s = 'baa' (second query)
aaa	s = 'aaa' (third query)

Sample Output

3

0

-1

Explanation

Query 1: "aaab"

Removing 'b' at index 3 results in a palindrome, so return 3.

Query 2: "baa"

Removing 'b' at index 0 results in a palindrome, so return 0.

Query 3: "aaa"

This string is already a palindrome, so return -1. Removing any one of the characters would result in a palindrome, but this test comes first.

Note: The custom checker logic for this challenge is available [here](#).