Full Name:	GUILHERME FERNANDES
Email:	contato@guifr.com.br
Test Name:	**Mock Test**
Taken On:	17 Jun 2022 19:55:42 IST
Time Taken:	29 min 52 sec/ 30 min
Invited by:	Ankush
Invited on:	17 Jun 2022 19:55:35 IST
Skills Score:	
Tags Score:	Algorithms 30/105 Core CS 30/105 Easy 30/105 Problem Solving 30/105 Strings 30/105 problem-solving 30/105

28.6%

30/105

scored in **Mock Test** in 29 min 52 sec on 17 Jun 2022 19:55:42 IST

Recruiter/Team Comments:

No Comments.

Question Description	Time Taken	Score	Status
Q1 **Palindrome Index** > **Coding**	30 min 38 sec	30/ 105	✓

QUESTION 1

Correct Answer

Score 30

Palindrome Index > Coding

Strings | Algorithms | Easy | problem-solving | Core CS | Problem Solving

QUESTION DESCRIPTION

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

$s = \text{"bcbc"}$

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.

CANDIDATE ANSWER

The candidate did not manually submit any code. The last compiled version has been auto-submitted and the score you see below is for the auto-submitted version.

Language used: **Java 8**

```
1  import java.io.*;
2  import java.math.*;
3  import java.security.*;
4  import java.text.*;
5  import java.util.*;
6  import java.util.concurrent.*;
7  import java.util.function.*;
8  import java.util.regex.*;
9  import java.util.stream.*;
10 import static java.util.stream.Collectors.joining;
```

```java
import static java.util.stream.Collectors.toList;



class Result {

    /*
     * Complete the 'palindromeIndex' function below.
     *
     * The function is expected to return an INTEGER.
     * The function accepts STRING s as parameter.
     */

    public static int palindromeIndex(String s) {
        System.out.println(s);
        int result = -1;

        StringBuilder order = new StringBuilder(s);
        StringBuilder reverse = new StringBuilder(s);
        reverse.reverse();

        if(order.toString().equals(reverse.toString())){
            return -1;
        }
        int mid = s.length() >> 1;

        boolean find = false;
        int max = Integer.MIN_VALUE;

        while(result < mid){
            result++;
            order = new StringBuilder(s);
            order.delete(result, result+1);
            reverse = new StringBuilder(order);
            reverse.reverse();
            if(order.toString().equals(reverse.toString())){
                find = true;
                max = Math.max(max,result);
            }
            int last = s.length() - result;
            order = new StringBuilder(s);
            order.delete(last, last+1);
            reverse = new StringBuilder(order);
            reverse.reverse();
            if(order.toString().equals(reverse.toString())){
                find = true;
                max = Math.max(max,last);
            }
        }

        return find ? max : -1;
    }

}

public class Solution {
    public static void main(String[] args) throws IOException {
        BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(System.in));
        BufferedWriter bufferedWriter = new BufferedWriter(new
FileWriter(System.getenv("OUTPUT_PATH")));

        int q = Integer.parseInt(bufferedReader.readLine().trim());

```

```
        IntStream.range(0, q).forEach(qItr -> {
            try {
                String s = bufferedReader.readLine();
                int result = Result.palindromeIndex(s);

                bufferedWriter.write(String.valueOf(result));
                bufferedWriter.newLine();
            } catch (IOException ex) {
                throw new RuntimeException(ex);
            }
        });

        bufferedReader.close();
        bufferedWriter.close();
    }
}
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 1	Easy	Sample case	Success	0	0.1373 sec	29.3 KB
Testcase 2	Medium	Hidden case	Success	5	0.1306 sec	30 KB
Testcase 3	Medium	Hidden case	Success	5	0.1706 sec	29.9 KB
Testcase 4	Medium	Hidden case	Success	5	0.1776 sec	29.8 KB
Testcase 5	Medium	Hidden case	Success	5	0.1374 sec	29.5 KB
Testcase 6	Medium	Hidden case	Terminated due to timeout	0	4.0376 sec	199 KB
Testcase 7	Medium	Hidden case	Terminated due to timeout	0	4.0661 sec	200 KB
Testcase 8	Medium	Hidden case	Terminated due to timeout	0	4.06 sec	199 KB
Testcase 9	Hard	Hidden case	Terminated due to timeout	0	4.0704 sec	200 KB
Testcase 10	Hard	Hidden case	Terminated due to timeout	0	4.225 sec	200 KB
Testcase 11	Hard	Hidden case	Terminated due to timeout	0	4.0296 sec	200 KB
Testcase 12	Hard	Hidden case	Success	10	0.1758 sec	29.5 KB
Testcase 13	Hard	Hidden case	Terminated due to timeout	0	4.0201 sec	201 KB
Testcase 14	Hard	Hidden case	Terminated due to timeout	0	4.0096 sec	199 KB
Testcase 15	Hard	Hidden case	Terminated due to timeout	0	4.0092 sec	200 KB

No Comments