Palindrome queries

Attempted by: 519 / Accuracy: 78% / Maximum Score: 30 / 7 Votes

Tag(s): Algorithms, String Algorithms, String Searching

You are given a string S that contains n lowercase alphabets. There are Q queries of the form $[L, R]$. Your task is to determine if every character in the range $[L, R]$ can be arranged in such a manner that a palindromic substring for that range can be formed. If it is possible to create the palindrome substring in the provided interval, then print **Possible**. Otherwise, print **Impossible**.

Note: The original string is not changed in the process.

Input format

- First line: Q denotes the number of queries
- Second lines: String S of n lowercase English letters
- Third line: Number of queries of the $[L, R]$ format

Output format

Print Q lines for each query. Print **Possible** if a palindrome substring is formed in the provided interval. Otherwise, print **Impossible**.

Constraints

$|s| \leq 10^5$
$|Q| \leq 10^5$
$|s| \leq 10^5$
$1 \leq L \leq R \leq n$

SAMPLE INPUT

```
4
abccab
1 3
2 3
3 3
3 5
```

SAMPLE OUTPUT

```
Impossible
Impossible
Possible
Possible
```

Explanation

In this case We can't make string palindrome if the interval is between [1, 3], [2, 3]. But we could make palindrome for the individual characters so it's **possible** for [3, 3].

Also for the case [3, 5] it's possible by shuffling the substring **cca** as **cac** so it can be formed as a palindrome.

Time Limit:	1.0 sec(s) for each input file.
Memory Limit:	256 MB
Source Limit:	1024 KB
Marking Scheme:	Marks are awarded when all the testcases pass.

Allowed Languages: Bash, C, C++, C++14, C++17, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift, Swift-4.1, TypeScript, Visual Basic

11

LIVE EVENTS

CODE EDITOR

Enter your code or Upload your code as file. Save Python 3 (python 3.5.2)

```
14        if len(L)==0:
15             return False
16        else:
17             if L==L[::-1]:
18                  return True
19             else:
20                  temp=permutations(L)
21                  for i in list(temp):
22                       if i ==i[::-1]:
23                            return True
24                  return False
25   Q=int(input())
26   S=str(input())
27   while (Q!=0) :
28        a=input()
29        b=a.split()
30        Li=list(int(i) for i in b)
31        Li[0]-=1
32        Lo=S[Li[0]:Li[1]]
33        Res=ispallin(Lo)
34        if Res is True:
35             print("Possible")
36        else:
37             print("Impossible")
38        Q-=1
39
```

37:20 vscode

Provide custom input

COMPILE & TEST SUBMIT

Log ID: 116988975 / Apr 19, 2020 12:50 PM IST

RESULT: Sample Test Cases Passed

Time (sec)	Memory (KiB)	Language
0.102977	64	Python 3

Input

```
4
abccab
1 3
2 3
3 3
3 5
```

Your Code's Output

```
Impossible
Impossible
Possible
Possible
```

Expected Correct Output

```
Impossible
Impossible
Possible
Possible
```

Compilation Log

Compiled successfully.

Execution Log

No execution log!

Need a hint?

Your Rating:

View all comments

PROGRAMMERS WHO SOLVED THIS PROBLEM ALSO SOLVED

Missing String
Attempted By: **209** / Accuracy: **19**
2 Votes

Predict The Road Sign - Here
Attempted By: **921** / Accuracy: **86**
1 Vote

Count Words In Given String
Attempted By: **94** / Accuracy: **42**
1 Vote