

ВШЭ АиСД 2021. Повторение C++, часть 2.

12 янв 2022, 18:04:17

старт: 10 сен 2021, 11:30:00

финиш: 19 сен 2021, 23:59:59

длительность: 9д. 12ч.

начало: 10 сен 2021, 11:30:00

конец: 19 сен 2021, 23:59:59

А. Негласный палиндром (0.2)

Ограничение времени	1 секунда
Ограничение памяти	64.0 Мб
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Возьмем произвольное слово и сделаем с ним следующую операцию: поменяем местами его первую согласную букву с последней согласной буквой, вторую согласную букву с предпоследней согласной буквой и т.д. Если после этой операции мы вновь получим исходное слово, то будем называть такое слово негласным палиндромом. Например, слова *sos*, *rare*, *rotor*, *gong*, *karaoke* являются негласными палиндромами.

Вам требуется написать программу, которая по данному слову определяет, является ли оно негласным палиндромом.

Формат ввода

Вводится одно слово. Длина строки ($|S| \leq 10^5$).

Формат вывода

Программа должна вывести YES, если введенное слово является негласным палиндромом, и NO в противном случае.

Пример 1

Ввод

tennete

Вывод

YES

Пример 2

Ввод

karaoke

Вывод

YES

Пример 3

Ввод

dream

Вывод

NO

```
1  /*Задача 1. Негласный палиндром*/
2
3  /*Возьмем произвольное слово и проделаем с ним следующую операцию: поменяем местами его первую
4  согласную букву с последней согласной буквой, вторую согласную букву с предпоследней согласной буквой и т.д.\
5  Если после этой операции мы вновь получим исходное слово, то будем называть такое слово негласным палиндромом.
6  Например, слова sos, rare, rotor, gong, karaoke являются негласными палиндромами.*/
7
8  #include <iostream>
9
10 bool isPalindrome(std::string str)
11 {
12     std::string consonantLetters;
13
14     for (int i = 0; i < str.length(); i++)
15     {
16         if (str[i] != 'a' && str[i] <= 'u' && str[i] != 'i' && str[i] != 'o' && str[i] != 'e' && str[i] != 'y')
17         {
18             consonantLetters += str[i];
19         }
20     }
21
22     int leftBoard = 0;
23     int rightBoard = consonantLetters.length() - 1;
24
25     while ((leftBoard != rightBoard && consonantLetters.length() % 2 == 1) || (leftBoard - 1 != rightBoard && consonantLetters.length() % 2 == 0))
26     {
27         if (consonantLetters[leftBoard++] != consonantLetters[rightBoard--])
28         {
29             return false;
30         }
31     }
32
33     return true;
34 }
35
36 int main()
37 {
38     //
```