**Uncle Grandpa and palindrome**

Uncle Grandpa is busy today (and tomorrow and the day after tomorrow), so this statement will be direct:

You are given a string consisting of only lower-case English letters. You have to determine if it’s possible to delete **at most** characters from the string to turn it into a palindrome.

A palindrome string is a sequence of characters which reads the same backward as forward, such as “madam”, “racecar”. An empty string, by definition, is also a palindrome.

## Input

The first line contains a single integer – the number of testcases

following lines, each line contains a string consisting of only lower-case English character and an integer

## Output

For each testcase, print if it’s possible to delete **at most** characters from the string to turn it into a palindrome and otherwise.

## Examples

|  |  |
| --- | --- |
| Input (palindrome1.in) | Output (palindrome1.out) |
| 3  pulalaup 2  madame 2  daevobvecd 1 | YES  YES  NO |

## Explaination:

For “pulalaup”, two character “a” can be deleted and turn the string to “pullup”, which is a palindrome.

For “madame”, the last “e” can be deleted and the string become “madam”

For “daevobvecd”, there exists no way to delete just 1 character to turn the string to a palindrome.

## Note:

1. A skeleton file has been given to help you. You should not create a new file or rename the file provided. You should develop your program using this skeleton file.
2. You are free to define your own helper methods and classes (or remove existing ones) if it is suitable but you must put all the new classes, if any, in the same skeleton file provided

## Skeleton File

You are given the skeleton file Palindrome.java. You should see the following contents when you open the file:

|  |
| --- |
| /\*\*  \* Name :  \* Matric. No :  \*/  import java.util.\*;  public class Palindrome {  private void run() {  }  public static void main(String args[]) {  Palindrome runner = new Palindrome();  runner.run();  }  } |