**Longest Palindrome in a String**

**Medium**

Given a string S, find the longest palindromic substring in S.**Substring of string S:** S[ i . . . . j ] where 0 ≤ i ≤ j < len(S)**. Palindrome string:** A string that reads the same backward. More formally, S is a palindrome if reverse(S) = S.**In case of conflict**, return the substring which occurs first ( with the least starting index).

**Example 1:**

**Input:**

S = "aaaabbaa"

**Output:** aabbaa

**Explanation**: The longest Palindromic

substring is "aabbaa".

**Example 2:**

**Input**:

S = "abc"

**Output:** a

**Explanation**: "a", "b" and "c" are the

longest palindromes with same length.

The result is the one with the least

starting index.

**Expected Time Complexity:**O(|S|2).  
**Expected Auxiliary Space:**O(1).

**Constraints:**  
1 ≤ |S| ≤ 103

**Company Tags**

[**Zoho**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Zoho) [**Accolite**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Accolite) [**Amazon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Amazon) [**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**Samsung**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Samsung) [**MakeMyTrip**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=MakeMyTrip) [**Visa**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Visa)[**Walmart**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Walmart) [**Google**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Google) [**Qualcomm**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Qualcomm) [**Groupon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Groupon)

**Topic Tags**

[**Strings**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Strings) [**Dynamic Programming**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Dynamic%20Programming) [**palindrome**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=palindrome) [**Data Structures**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Data%20Structures) [**Algorithms**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Algorithms)

//{ Driver Code Starts

//Initial Template for Java

import java.io.\*;

import java.util.\*;

class CodingMaxima

{

public static void main(String args[])throws IOException

{

BufferedReader read = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(read.readLine());

while(t-- > 0)

{

String S = read.readLine();

Solution ob = new Solution();

System.out.println(ob.longestPalin(S));

}

}

}

// } Driver Code Ends

//User function Template for Java

class Solution{

static String longestPalin(String S){

String ans="";

for(int i=0;i<S.length();i++){

String oddP=find(S, i, i);

String evenP=find(S, i, i+1);

oddP=oddP.length()>evenP.length()?oddP: evenP;

if(oddP.length()>ans.length()){

ans=oddP;

}

}

return ans;

}

private static String find(String s, int i, int j){

while(i>=0 && j<s.length() && s.charAt(i)==s.charAt(j)){

i--;

j++;

}

return s.substring(i+1, j);

}

}