**★★★☆☆**

**題組：Problem Set Archive** **with Online Judge**

**題號：11151: Longest Palindrome**

**解題者：洪偉盛**

**解題日期：2022年4月7日**

**題意：**

給定一個字串，判斷字串中最長的回文字串

**題意範例：**

ADAM

ABDAM

MADAM

3

3

5

**解法：**

首先，把字串反轉reverse(因為題目要求的是回文字 串)然後再找出”輸入字串”跟”其反轉字串”兩字串的最長共同子字串就是答案了

以下是LCS的DP公式:

宣告一個二維陣列DP[][]來記錄

 DP[i][j] = DP[i-1][j-1]+1 (X[i]==Y[j])

 DP[i][j] = max(DP[i][j-1],DP[i-1][j] (X[i]=Y[j])

EX:

 輸入字串:ABDAM

 反轉字串:MADBA

X:ABDAM

Y:MADBA

 M!=A

那麼X和Y的最長共同子字串就會是

ABDAM 、 MADB 的LCS 跟 ABDA、MADBA 的LCS
兩組中取大的。

EX:

 輸入字串:ABDA

 反轉字串:ADBA

X:ABDA

Y:ADBA

 A=A

那麼X和Y的最長共同子字串就會是

(ABD 、 ADB 的LCS)+A

(AB)+A => ABA

**解法範例：**

若有範例，則列出。若沒有，則填寫「無」。

**討論：**

反轉字串=>可以#include <algorithm> 中的

reverse (string.begin(),string.end())

**程式：**

#include <iostream>

#include <cstdlib>

#include <algorithm>

#include <cstring>

#include <cmath>

#include <string>

using namespace std;

int main(){

 int time;

 cin>>time;

 cin.get();

 while(time>0){

 string str;

 getline(cin,str);

 string strc;

 strc = str;

 reverse(str.begin(),str.end());

 int len = str.length();

 int dp[len+1][len+1];

 memset(dp,false,sizeof(dp));

 for(int i=1;i<=len;i++){

 for(int j=1;j<=len;j++){

 if(strc[i-1] == str[j-1]){

 dp[i][j] = dp[i-1][j-1] + 1;

 }else{

 dp[i][j] = max(dp[i-1][j],dp[i][j-1]);

 }

 }

 }

 /\*for(int i=0;i<=len;i++){

 for(int j=0;j<=len;j++){

 cout<<dp[i][j];

 }

 cout<<endl;

 }\*/

 cout<<dp[len][len]<<endl;

 time--;

 }

 return 0;

}