UVA272

題目

TEX is a typesetting language developed by Donald Knuth. It takes source text together with a few typesetting instructions and produces, one hopes, a beautiful document. Beautiful documents use " and " to delimit quotations, rather than the mundane " which is what is provided by most keyboards. Keyboards typically do not have an oriented double-quote, but they do have a left-single-quote ` and a right-single-quote '. Check your keyboard now to locate the left-single-quote key ` (sometimes called the "backquote key") and the right-single-quote key ' (sometimes called the "apostrophe" or just "quote"). Be careful not to confuse the left-single-quote ` with the "backslash" key \. TEX lets the user type two left-single-quotes `` to create a left-double-quote " and two right-single-quotes '' to create a right-double-quote ". Most typists, however, are accustomed to delimiting their quotations with the un-oriented double-quote ".

If the source contained

"To be or not to be," quoth the bard, "that is the question."

then the typeset document produced by TEX would not contain the desired form:

"To be or not to be," quoth the bard, "that is the question."

In order to produce the desired form, the source file must contain the sequence:

``To be or not to be,'' quoth the bard, ``that is the question.''

You are to write a program which converts text containing double-quote (") characters into text that is identical except that double-quotes have been replaced by the two-character sequences required by TEX for delimiting quotations with oriented double-quotes. The double-quote (") characters should be replaced appropriately by either `` if the " opens a quotation and by '' if the " closes a quotation. Notice that the question of nested quotations does not arise: The first " must be replaced by ``, the next by '', the next by ``, the next by '', the next by ``, the next by '', and so on.

題目

TEX is a typesetting language developed by Donald Knuth. It takes source text together with a few typesetting instructions and produces, one hopes, a beautiful document. Beautiful documents use " and " to delimit quotations, rather than the mundane " which is what is provided by most keyboards. Keyboards typically do not have an oriented double-quote, but they do have a left-single-quote ` and a right-single-quote '. Check your keyboard now to locate the left-single-quote key ` (sometimes called the "backquote key") and the right-single-quote key ' (sometimes called the "apostrophe" or just "quote"). Be careful not to confuse the left-single-quote ` with the "backslash" key \. TEX lets the user type two left-single-quotes `` to create a left-double-quote " and two right-single-quotes '' to create a right-double-quote ". Most typists, however, are accustomed to delimiting their quotations with the un-oriented double-quote ".

TeX 是一種由 Donald Knuth 所發展出的一套文書排版軟體。這套軟體可以將原始文件檔加上一些像字型等型態後，轉成一份很漂亮的文件。而一份漂亮的文件是需要用 `` 和 " 來把別人說的話給「引」出來，而不是用大部份鍵盤上有的 " 。雖然鍵盤裡通常不會有這種有方向的雙引號鍵，不過上面有左單引號 `（有人叫 backquote ），和右單引號 '（有人叫 apostrophe 或 quote ）。你可以在你的鍵盤上找一下，不過要小心不要將 ` 與 \（ backslash 鍵）搞混了。而在 TeX 裡，使用者可以輸入兩個左單引號 `` 來構成一個左雙引號 ``，或者是兩個右單引號 '' 來構造一個右單引號 ''，不過呢，通常大家打字時都很習慣用普通的雙引號 " 來引述別人的話。

題目

If the source contained

"To be or not to be," quoth the bard, "that is the question."

then the typeset document produced by TEX would not contain the desired form:

"To be or not to be," quoth the bard, "that is the question."

In order to produce the desired form, the source file must contain the sequence:

``To be or not to be,'' quoth the bard, ``that is the question.''

如果原始文件檔內容是：
"To be or not to be," quoth the bard, "that is the question."
則 TeX 作出來的文件並不會是我們所想要的：
``To be or not to be," quoth the bard, ``that is the question."
為了要得到上面的文件，我們需要把原始文件變成這個樣子：
``To be or not to be,'' quoth the bard, ``that is the question.''

題目

You are to write a program which converts text containing double-quote (") characters into text that is identical except that double-quotes have been replaced by the two-character sequences required by TEX for delimiting quotations with oriented double-quotes. The double-quote (") characters should be replaced appropriately by either `` if the " opens a quotation and by '' if the " closes a quotation. Notice that the question of nested quotations does not arise: The first " must be replaced by ``, the next by ", the next by ``, the next by ", the next by ``, the next by ", and so on.

你現在必須要寫一個程式,將普通的雙引號 ("),轉成有方向性的雙引號,而其它文字則不變。而在把普通的雙引號換掉的時候,要特別注意,當要開始引述一句話時要用 ``,而結束引述時要用 "。不用擔心會有多層巢狀引號的情形,也就是第一個引號一定是用 `` 來代替,再來用 ",然後用 ``,接著用 ",依此類推。

題目

TeX 是一種由 Donald Knuth 所發展出的一套文書排版軟體。這套軟體可以將原始文件檔加上一些像字型等型態後，轉成一份很漂亮的文件。而一份漂亮的文件是需要用 `` 和 " 來把別人說的話給「引」出來，而不是用大部份鍵盤上有的 " 。雖然鍵盤裡通常不會有這種有方向的雙引號鍵，不過上面有左單引號 `（有人叫 backquote ），和右單引號 '（有人叫 apostrophe 或 quote ）。你可以在你的鍵盤上找一下，不過要小心不要將 ` 與 \（ backslash 鍵）搞混了。而在 TeX 裡，使用者可以輸入兩個左單引號 `` 來構成一個左雙引號 `` ，或者是兩個右單引號 '' 來構造一個右單引號 '' ，不過呢，通常大家打字時都很習慣用普通的雙引號 " 來引述別人的話。

如果原始文件檔內容是：

"To be or not to be," quoth the bard, "that is the question."

則 TeX 作出來的文件並不會是我們所想要的：

``To be or not to be," quoth the bard, ``that is the question."

為了要得到上面的文件，我們需要把原始文件變成這個樣子：

``To be or not to be,'' quoth the bard, ``that is the question.''

你現在必須要寫一個程式，將普通的雙引號（ " ），轉成有方向性的雙引號，而其它文字則不變。而在把普通的雙引號換掉的時候，要特別注意，當要開始引述一句話時要用 `` ，而結束引述時要用 '' 。不用擔心會有多層巢狀引號的情形，也就是第一個引號一定是用 `` 來代替，再來用 '' ，然後用 `` ，接著用 '' ，依此類推。

輸入與輸出

Input：
Input will consist of several lines of text containing an even number of double-quote (") characters.
Input is ended with an end-of-file character.

Output：
The text must be output exactly as it was input except that:

▪ the first " in each pair is replaced by two ` characters: `` and

▪ the second " in each pair is replaced by two ' characters: ''.

輸入：輸入是若干列的文字，其中有偶數個雙引號（"），以 end-of-file 做結束。

輸出：輸出的文字必須和輸入的一模一樣，除了：
■ 每一組雙引號的第一個 " 必須用兩個 ` 字元（就是 ``）來代替
■ 每一組雙引號的第二個 " 必須用兩個 ' 字元（就是 ''）來代替。

範例測資

Input：
"To be or not to be," quoth the Bard, "that
is the question".
The programming contestant replied: "I must
disagree.
To `C' or not to `C', that is The Question!"

Output：
``To be or not to be,'' quoth the Bard, ``that
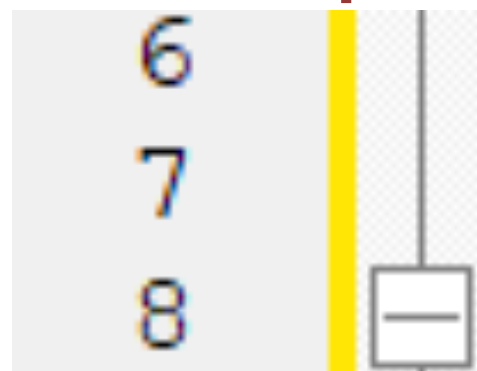is the question''.
The programming contestant replied: ``I must
disagree.
To `C' or not to `C', that is The Question!''

程式碼說明

Step 1：輸入測資

```
6        string s;
7        int time=1;
8        while(getline(cin,s)){
```

已宣告變數	註解
s	文字
time	第一個或第二個雙引號

程式碼說明

Step 2：判斷是否為雙引號並判別是第一個還是第二個雙引號

```
9     for(char i:s){
10        if(i=='"')
11            if(time++%2==1)
12                cout<<"``";
13            else
14                cout<<"''";
15        else
16            cout<<i;
17    }
```

已宣告變數	註解
s	文字
time	第一個或第二個雙引號

完整程式碼

```cpp
#include<iostream>
#include<string>

using namespace std;
int main(){
    string s;
    int time=1;
    while(getline(cin,s)){
        for(char i:s){
            if(i=='"')
                if(time++%2==1)
                    cout<<"``";
                else
                    cout<<"''";
            else
                cout<<i;
        }
        cout<<endl;
    }
}
```

資料來源

英文題目：

https://vjudge.net/problem/UVA-272

中文翻譯：

https://zerojudge.tw/ShowProblem?problemid=c007

Thanks!