源程序

```
#include<fstream>
#include<iostream>
using namespace std;
void pretreatment(char filename[],char Buf[]){

    // 读取原文件
    ifstream cinf(filename,ios::in);

    int i = 0;
    char c;
    bool in_comment=false;

    cout<<"<源程序>"<<endl;

    // 处理{}中间的注释部分
    // void *p = cinf.read(&c,sizeof(char));
    while(cinf.read(&c,sizeof(char))){
        cout<<c;
        switch(in_comment){
            case false:
                if(c=='{') in_comment=true;
                else{
                    // 替换Tab和换行符为空格
                    if(c==0x9||c==0xa) c = 0x20;
                    Buf[i++] = c;
                }
                break;
            case true:
                if(c=='}') in_comment=false;
        }
        // p = cinf.read(&c,sizeof(char));
    }
    // 在结尾加入#作为结束
    Buf[i] = '#';
}
int main(){
    char Buf[4048] = {'\0'};
    pretreatment("source.txt",Buf);
    cout<<"<预处理结果>"<<endl;
    cout<<Buf<<endl;
}
```

处理结果

```
[Running] cd
"f:\Notes\BasicNotes\Notesys\ComputerScience\CompilationTheory\Project
s\" && g++ lexer_example.cpp -o lexer_example &&
"f:\Notes\BasicNotes\Notesys\ComputerScience\CompilationTheory\Project
s\"lexer_example
lexer_example.cpp: In function 'int main()':
lexer_example.cpp:35:34: warning: ISO C++ forbids converting a string
constant to 'char*' [-Wwrite-strings]
     pretreatment("source.txt",Buf);
                                  ^
<源程序>
while(p){
        cout<<c;
        switch(in_comment){
            case false:
                if(c=='{') in_comment=true;
                else{
                    if(c==0x9||c==0xa) c = 0x20;
                    Buf[i++] = c;
                }
                break;
            case true:
                if(c=='}') in_comment=false;
        }
        p = (char *)cinf.read(&c,sizeof(char));
    }
    Buf[i] = '#';
}<预处理结果>
while(p)                 break;             case true:
if(c=='}') in_comment=false;         }         p = (char
*)cinf.read(&c,sizeof(char));     }     Buf[i] = '#'; }#

[Done] exited with code=0 in 1.159 seconds
```