

ch5 状压动态规划案例-消除字符串

小明喜欢中心对称的字符串，即回文字符串。现在小明手里有一个字符串 S ，小明每次都会进行这样的操作：从 S 中挑选一个回文的子序列，将其从字符串 S 中去除，剩下的字符重组成新的字符串 S 。小明想知道，最少可以进行多少次操作，可以消除整个字符串。

输入格式：输入一行。输入一个字符串 S ($1 \leq \text{length}(S) \leq 16$)，字符串均由小写字母组成。

输出格式：输出一行，输出一个整数，表示消除整个字符串需要的最少操作次数。

样例输入

abaccba

样例输出

2

参考代码：

```
#include<bits/stdc++.h>
using namespace std;
int dp[(1<<16)+10]={0};
string s;
const int inf =0x3f3f3f3f;
int check(int j){
    string s1="",s2="";
    int pos=0;
    while(j){
        if(j&1){
            s1+=s[pos];
        }
        pos++;
        j>>=1;
    }
    s2=s1;
    reverse(s2.begin(),s2.end());
    return s1==s2;
}
int main(){
    cin>>s;
    int t=s.size();
    for(int i=1;i<(1<<t);i++){
        if(check(i)){
            dp[i]=1;
        }else{
            dp[i]=inf;
            for(int j=i;j=(j-1)&i){
                dp[i]=min(dp[i],dp[j]+dp[j^i]);
            }
        }
    }
}
```

```
    cout<<dp[(1<<t)-1]<<endl;  
    return 0;  
}
```