[**新一代字典树模板**](http://www.cppblog.com/notonlysuccess/archive/2009/02/10/73375.html)

<http://acm.hdu.edu.cn/showproblem.php?pid=1251>

http://www.cppblog.com/Images/OutliningIndicators/None.gif#include<stdio.h>  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifstruct treehttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    bool isword;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int child[26],cnt;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}T[1000001];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifchar word[11];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint L = 1,ans;//树的总长  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid Ins(char \*a,int k,int idx)//k代表单词长度，也是深度，idx代表树的ID  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(!a[k])//这个单词建完  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        return ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif   if(T[idx].child[a[k] - 'a'])//找到这个单词，继续往下找  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T[ T[idx].child[ a[k] -'a'] ].cnt ++ ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Ins(a,k+1,T[idx].child[a[k]-'a']);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    else  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T[idx].child[a[k]-'a'] = ++L;//树的总长++  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T[L].isword = false;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T[L].cnt ++;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Ins(a,k+1,L);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid find(char \*a ,int k,int idx)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(!a[k])  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        ans = T[idx].cnt;//全部找完，返回  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        return ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(T[idx].child[a[k]-'a'] == 0)//找不到字母了，一定不是前缀  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        return ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    find(a,k+1,T[idx].child[a[k]-'a']);//继续找下一个字母  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint main()  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    T[1].isword = false;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    while (gets(word),word[0])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Ins(word,0,1);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    while (gets(word))  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        ans = 0;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        find(word,0,1);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        printf("%d\n",ans);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    return 0;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}