

trie(字典树)

```
#include <bits/stdc++.h>
using namespace std;
char s[10010];
int nex[500010][30], n, cnt=0;
bool b[500010][110];

inline void insert(int x){
    scanf("%s", s+1);
    int len=strlen(s+1);
    int now=0;
    for(int i=1;i<=len;i++){
        int p=s[i]-'a';
        if(!nex[now][p]) nex[now][p]=++cnt;
        now=nex[now][p];
    }
    b[now][x]=1;
}

inline void check(){
    scanf("%s", s+1);
    int len=strlen(s+1);
    int now=0, flag=1;
    for(int i=1;i<=len;i++){
        int p=s[i]-'a';
        if(!nex[now][p]) { //如果在Trie树中没有当前的字符，就可以直接break掉了
            flag=0;
            break;
        }
        now=nex[now][p]; //否则就更新位置
    }
    if(flag){
        for(int i=1;i<=n;i++) { //题面上说按字典序输出
            if(b[now][i]) cout << i; //输出在哪些句子中出现过
        }
    }
    cout << endl;
}

int main(){
    cin >> n;
    for(int i=1;i<=n;i++){
        int x;
        cin >> x;
        for(int j=1;j<=x;j++){ //一个单词一个单词的插入Trie树里
            insert(j);
        }
    }
    int m;
    cin >> m;
    for(int i=1;i<=m;i++){
        check();
    }
}
```

```
    return 0;  
}
```