



正在挑战一个CrackMe 的你，把需要填写的前面几位密码都正确猜出了，可是这最后一位密码，好像藏得有点深。CrackMe 的作者还挑衅般的在里面藏了一个.tar.gz 文件，解压缩出来，里面写道  
你要的最后一个字符就在下面这个字符串里，这个字符是下面整个字符串中第一个只出现一次的字符。(比如，串是 abaccdeff，那么正确字符就是 b 了)然而下面给出来的字符串好像太长太长了，单靠人力完全无法找出来。  
于是，你需要写一个程序代劳了。输入文件体积较大，请使用一些快速的输入输出手段，不推荐使用 cin/cout，对 Java 并不推荐使用Scanner 直接读写。以下

是标程：

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4
5  typedef long long ll;
6
7  const int MAXN=1000000;
8
9  int idx[256];
10 int T;
11 char str[MAXN+5];
12
13 int main() {
14     for(scanf("%d",&T);T--;) {
15         scanf("%s",str+1);
16         memset(idx,0,sizeof(idx));
17         for(int i=1;str[i];i++) {
18             if(idx[str[i]]==0) {
19                 idx[str[i]]=i;
```



```
20         }else{
21             idx[str[i]]=-1;
22         }
23     }
24     char ans='\0';
25     for(int i=0x21;i<0x7F;i++){
26         if(idx[i]>0){
27             if(ans==0||idx[ans]>idx[i]) ans=i;
28         }
29     }
30     printf("%c\n",ans);
31 }
32 return 0;
33 }
```

360 员工桂最近申请了一个长假，一个人背着包出去自助游了。

路上，他经过了一个小镇，发现小镇的人们都围在一棵树下争吵。桂上前询问情况，得知小镇的人们正缺一个镇长，他们希望能选一个知名又公正的镇长，即，大家希望能选出一个人，所有人都认识他，但同时他不认识镇上除自己以外的其他人（在此，我们默认每个人自己认识自己）。可是小镇里的人太多了，一下子大家谁也说服不了谁。

“这简单啊。”桂表示。于是他一下子统计出来了镇上人们相互之间的认识关系，并且一下子找到了合适的镇长人选。

现在你手上也拿到了这样一份认识关系的清单。其中上面给出的认识关系是单向的，即，**A** 认识 **B** 与 **B** 认识 **A** 是相互独立的，只给出 **A** 认识 **B** 就不能认为 **B** 认识 **A**，例如，我认识你，你不一定认识我。而且，这里的认识关系也不具有传递性，即，**A** 认识 **B**，**B** 认识 **C**，但这不代表 **A** 认识 **C**。同时，为了方便处理，这份清单中，镇上的 **N** 个人依次编号为 **1** 到 **N**。你能否像桂一样快速找到合适的镇长人选呢？

```
1 #include <iostream>
2 using namespace std;
```



```
3
4 int main()
5 {
6     int T;
7     cin>>T;
8     int n,m;
9     int a,b;
10    while(T-->0)
11    {
12        cin>>n>>m;
13        int *man_a =new int[n];
14        int *man_b =new int[n];
15        int zhen=0;
16        for(int i=0; i<n; ++i)man_b[i]=man_a[i]=0;
17        while(m-->0)
18        {
19            cin>>a>>b;
20            if(a!=b)
21            {
22                man_a[a-1]++;
23                man_b[b-1]++;
24            }
25        }
26        for(int i=0; i<n; ++i)
27        {
28            if(man_a[i] == 0 && man_b[i]==n-1)
```



```
29         {
30             zhen = i+1;
31             break;
32         }
33     }
34     if(zhen>0)
35     {
36         cout<<1<<endl;
37         cout<<zhen;
38     }
39     else cout<<0<<endl;
40     cout<<endl;
41     delete []man_a;
42     delete []man_b;
43 }
44 return 0;
45 }
```



icebear.me

白熊事务所致力为准备求职的小伙伴提供优质的资料礼包和高效的求职工具。礼包包括**互联网、金融等行业的求职攻略**；**PPT模板**；**PS技巧**；**考研资料**等。

微信扫码关注：**白熊事务所**，获取更多资料礼包。

登陆官网：**www.icebear.me**，教你如何**一键搞定名企网申**。