

Problema H

O Jogo

Todas as pessoas do mundo estão jogando O Jogo, mesmo que não saibam disso. Aliás, as pessoas que não sabem dele, estão ganhando! O Jogo é um jogo muito simples, tendo apenas duas regras:

- Toda vez que alguém pensa no jogo, perde.
- Toda vez que alguém perde, deve alertar todos a sua volta que ele perdeu.

Bom, você acaba de perder O Jogo... Mas não se preocupe, pois assim que esquecer-lo, voltará a jogá-lo!

Obviamente, um jogo como esse nunca terá um vencedor já que as regras somente definem um critério de perda. Contudo, você como funcionário da maior rede de micro-blogging do mundo e inconformado de não haver um vencedor neste jogo resolveu criar um prêmio para o usuário da sua rede que menos pensou no jogo.

Para alcançar seu objetivo, você terá que analisar o histórico de todos os eventos que aconteceram na rede. O funcionamento da sua rede é baseado numa arquitetura publish/subscribe e conta com as seguintes três primitivas:

- SUBSCRIBE A, B - usuário A passa a seguir B, ou seja, recebe mensagens dele
- UNSUBSCRIBE A, B - usuário A deixa de seguir B
- PUBLISH A - usuário A publica uma mensagem a todos que o seguem

As mensagens de PUBLISH já foram filtradas e só aparecem no histórico aquelas que anunciam O Jogo, fazendo com que todos que a recebem lembrem-se dele. Parece que nem todas as pessoas obedecem as regras do jogo, sendo comum que alguém receba uma mensagem falando sobre O Jogo e não avise aos seus amigos que lembrou do jogo. Contudo, tanto quem publica uma mensagem quanto quem a recebe lembra do jogo.

O seu histórico é dado em texto puro com uma primitiva por linha. Em toda linha, a primitiva é precedida por um inteiro, marcando o dia no qual a primitiva foi trocada. Nem todo dia há troca de mensagens. Como exemplo, é dado o trecho abaixo:

```
123 SUBSCRIBE alice, bob
124 PUBLISH bob
124 PUBLISH alice
126 PUBLISH alice
```

No exemplo acima, no dia 123, alice passa a seguir bob sem que bob siga alice. No dia seguinte, bob lembra do jogo e acaba avisando alice através de uma mensagem. Nesse mesmo dia, alice publica uma mensagem sobre o jogo, lembrando também seus seguidores. Já no dia 126, somente alice lembra do jogo. Neste caso, alice lembrou do jogo dois dias e bob apenas um.

Como sua rede possui muitos usuários, você não sabe exatamente quantos usuários aparecem no histórico e deve, portanto, estar preparado para suportar vários. Cada usuário é representado por uma string de no máximo 15 caracteres, somente letras minúsculas são usadas.

Dado o histórico da rede em ordem cronológica, inclusive em relação às mensagens trocadas no mesmo dia, descubra qual usuário lembrou menos do jogo, ou seja, quem passou mais dias, não necessariamente consecutivos, sem lembrar do jogo.

Entrada

A entrada contém vários casos de teste. Cada caso de teste começa com um número $1 \leq n \leq 10.000$ que representa a quantidade de mensagens no histórico. Essa linha é seguida de n linhas contendo primitivas como descrito anteriormente. O número de usuários distintos em um histórico será no máximo 10.000.

A entrada termina com um caso de teste contendo um valor $n = 0$ e não deve ser processado.

Saída

A saída deve conter uma linha para cada caso de teste. Para cada caso de teste, escreva na saída o usuário que pensou no jogo por menos dias. Em caso de empate, imprima todos os usuários em ordem alfabética e separados por um espaço.

Exemplo de entrada 1	Exemplo de saída 1
4 1 SUBSCRIBE ana, carlos 1 SUBSCRIBE ana, sabrina 2 PUBLISH sabrina 3 SUBSCRIBE carlos, ana 6 1 SUBSCRIBE antonio, barbara 2 SUBSCRIBE barbara, cristina 3 SUBSCRIBE cristina, antonio 4 PUBLISH antonio 4 PUBLISH barbara 4 PUBLISH cristina 0	carlos antonio barbara cristina