

Orkut

Arquivo fonte: orkut.c, orkut.cc, orkut.cpp ou orkut.pas

Larissa acaba de entrar para o Orkut, um site na internet que permite que as pessoas se reúnam em comunidades e grupos de amigos. Como ela acabou de se registrar, ela ainda não possui muitos amigos na sua lista de contatos. Após fazer uma pesquisa, ela descobriu que os seus antigos amigos de escola (que adoravam mexer com computadores) também fazem parte do Orkut. Larissa então decidiu chamá-los para serem seus amigos virtuais. Porém, eles resolveram brincar com a Larissa, e cada um deles só vai aceitar o pedido de Larissa quando ela já for amiga virtual de alguns dos outros amigos do grupo. Assim, para conseguir ter todos os seus antigos amigos de escola na sua lista de amigos do Orkut, ela deve cumprir as exigências de cada um deles.

1. Tarefa

Larissa acha que pode encontrar uma seqüência de nomes dos amigos, de modo que se ela pedir a cada um deles para ser sua amiga no Orkut, obedecendo a seqüência, todas as exigências serão cumpridas e todos eles irão aceitar o seu pedido. Larissa precisa da sua ajuda para resolver esse problema de forma rápida. A sua tarefa é escrever um programa para encontrar uma seqüência de nomes que resolva o problema, ou dizer que não é possível encontrar tal seqüência.

2. Entrada

A entrada é composta de vários conjuntos de teste. A primeira linha de um conjunto de teste contém um inteiro N que indica o número de antigos amigos da Larissa ($1 \leq N \leq 30$). A linha seguinte irá conter N nomes de amigos, separados por espaço em branco. Cada nome não terá mais de 15 letras, e serão todos distintos. Nas próximas N linhas serão indicadas as exigências que a Larissa deve cumprir. Cada linha descreve a exigência de um amigo e começará com o nome desse amigo, seguido de um número M ($0 \leq M \leq N-1$), que indica o número de pessoas que aquele amigo quer que a Larissa seja amiga antes, e seguido pelos M nomes de amigos (cada item na linha separado por espaço em branco). O final da entrada é indicado por $N = 0$.

Exemplo de Entrada

```
5
Joao Maria Tadeu Jose Ricardo
Joao 2 Maria Ricardo
Maria 1 Tadeu
Jose 1 Joao
Tadeu 0
Ricardo 0
3
Joao Maria Renata
Maria 1 Joao
Joao 1 Renata
Renata 1 Maria
0
```

3. Saída

Para cada conjunto de teste seu programa deve produzir três linhas na saída. A primeira linha deverá conter um identificador do conjunto de teste, no formato “Teste n ”, onde n é numerado seqüencialmente a partir de 1. A segunda linha deve conter a seqüência de nomes de amigos (cada nome seguido de um espaço em branco) que resolve o problema da Larissa, ou a palavra “impossível”, quando não houver uma seqüência possível (note a ausência de acentuação). Se existir mais de uma seqüência de amigos que resolve o problema, imprima qualquer uma delas (mas apenas uma). A terceira linha deverá ser deixada em branco. A grafia mostrada no Exemplo de Saída abaixo deverá ser seguida rigorosamente.

Exemplo de Saída

```
Teste 1
Ricardo Tadeu Maria Joao Jose
```

```
Teste 2
impossivel
```

(esta saída corresponde ao exemplo de entrada acima)

4. Restrições

$0 \leq N \leq 30$ ($N = 0$ apenas para indicar o fim da entrada)

$0 \leq M \leq N - 1$

Cada nome de amigo terá no máximo 15 letras