

Gostar ou não gostar de SPAM?

Hoje em dia, infelizmente, mensagens de SPAM estão ficando mais e mais comuns. Algumas delas têm efeito de multiplicação, já que pedem que repasses para teus amigos. Algumas mensagens de SPAM desejam boa sorte, outras, prometem que tornar-se-á rico, e outras apenas lhe lembram o quão importante é contar a seus amigos o quanto você se importa com suas amizades. Eis um exemplo de SPAM:

De: Alice
Para: Bob, Mary, Julia, Paul

Olá, este é um e-mail de boa sorte. Desejo que se torne milionário, mas isso depende de você. Se você:

* mandar esse e-mail para 10 pessoas você se ficará milionário

* mandar esse e-mail para 5 ou mais pessoas você ficará rico

* mandar esse e-mail para menos de 5 pessoas você será pobre

Como eu disse, depende de você. Escreve seu e-mail e seja rico! :-)

Alice

Pessoas normalmente reagem de dois modos diferentes quando recebem SPAM:

- Eles descartam a mensagem imediatamente sem ao menos lê-la (eles odeiam SPAM);
- Eles encaminham a mensagem para todos que conhecem (eles amam SPAM).

Para esse problema, nós assumiremos que todos amam SPAM, mas ninguém manda a mesma mensagem duas vezes. Cada mensagem de SPAM tem um efeito baseado no número de amigos que você manda as mensagens. Por exemplo: uma mensagem de SPAM poderá lhe dizer que ficarás pobre se mandares a mensagem para 5 amigos, mas serás rico se mandares para 10, e serás o mais rico do mundo se mandares para 20 amigos, e assim vai.

Consideraremos apenas as mensagens de SPAM similares ao exemplo acima. Mais especificamente, uma mensagem de SPAM definirá dois valores limiares T_1 e T_2 e três atribuidores A_1 , A_2 e A_3 . Uma pessoa adquire um dos três valores dependendo do número de mensagens enviadas para aquele SPAM específico. Se a pessoa encaminha T mensagens e $T < T_1$ então o atributo dele/dela é A_1 , se $T_1 \leq T < T_2$ então o atributo dele/dela é A_2 , caso contrário o atributo dela é A_3 .

Lhe será dado os nomes de um grupo de pessoas, e para cada pessoa do grupo, um conjunto de amigos que ele/ela conhece o endereço de e-mail. Lhe será dado um conjunto distinto de mensagens de SPAM, e para cada mensagem de SPAM ela limita valores e atributos, e a informação sobre cada pessoa que iniciou a proliferação do SPAM. Você tem que escrever um programa que determine, para cada pessoa no grupo dado, quais atributos ele/ela adquiriu, baseado no SPAM encaminhado.

Você assumirá que o originador do SPAM tenha pelo menos um amigo (em outras palavras, ele/ela enviará pelo menos uma mensagem), e a pessoa não mandará para si mesma.

Entrada

Seu programa deverá processar vários casos de teste. A primeira linha do caso de teste contém um inteiro N indicando o número de pessoas no grupo ($2 \leq N \leq 20$). Na entrada a pessoa é identificada por um inteiro de 1 a N . As N linhas seguintes contêm cada, uma lista de amigos de cada pessoa (a linha i contém a lista de amigos da pessoa i). A lista de amigos da pessoa i descreve os amigos que a pessoa i conhece o endereço de e-mail, e consiste de uma lista de inteiros F_i ($1 \leq F_i \leq N$, $F_i \neq i$) terminada pelo valor 0 (zero). Seguindo a lista de amigos vem a descrição das mensagens de SPAM (terá no máximo 100 mensagens). Cada descrição aparece em uma linha diferente. A descrição consiste de um inteiro P identificando a pessoa originadora do SPAM ($2 \leq P \leq N$); dois inteiros T_1 e T_2 representando os valores limites; e os três atributos A_1 , A_2 e A_3 (cada atributo é uma palavra de não mais que vinte letras). A lista de SPAM acaba com uma linha contendo apenas o valor 0 (zero). As N linhas contêm cada, um nome, que é uma palavra apenas, com não mais que vinte letras. O nome na linha de i é o nome da pessoa número i . O fim da

entrada é indicado por **N** = 0.

Saída

Para cada caso de teste seu programa deverá exibir uma lista de nomes seguida dos atributos que adquiriram. Seu programa deve escrever os nomes das pessoas na ordem em que aparecerem na entrada, seguido de ':' e de um espaço, seguido de seus atributos de acordo com os SPAM que enviaram. Os atributos devem ser escritos na ordem que apareceram na entrada; cada atributo deve ser seguido de espaço.

Exemplo de entrada

```
5
2 3 0
1 3 5 4 0
5 0
0
4 1 0
1 2 4 poor rich millionaire
5 3 10 sad normal happy
0
Bob
Paul
Mary
Alice
Julia
6
2 0
1 3 0
1 2 4 0
1 2 3 5 0
1 2 3 4 0
1 3 4 0
1 2 4 red green blue
1 2 4 dumb normal smart
6 3 5 ugly bad good
0
Peter
Paul
Victoria
John
Julia
Anne
0 0
```

Exemplo de saída

```
Bob: rich sad
Paul: millionaire normal
Mary: poor sad
Alice: poor sad
Julia: rich sad
Peter: red dumb ugly
```

Paul: green normal ugly
Victoria: green normal bad
John: blue smart bad
Julia: blue smart bad
Anne: red dumb bad