

Meu Avô é Famoso

Por Ricardo Anido  Brasil

Timelimit: 3

A família toda ficou excitada pela novidade. Todos sabiam que o meu avô tinha sido um excelente jogador de bridge por décadas, mas quando foi anunciado que ele estaria no Guinness Book, o livro dos recordes, como o jogador de bridge de maior sucesso de todos os tempos, wow, aquilo foi surpreendente.

A Associação Internacional de Bridge (AIB) tem mantido, por diversos anos, um ranking semanal dos melhores jogadores do mundo. Considerando que cada aparição em um ranking semanal constitui um ponto para o jogador, meu avô foi nominado o melhor jogador de todos os tempos porque ele conseguiu o maior número de pontos.

Tendo muitos amigos que também estavam competindo com ele, meu avô está extremamente curioso para saber que jogador(es) ficou(aram) com o segundo lugar. Ele precisa de um programa, o qual, dada uma lista com os ranking semanais, descubra que jogador(es) ficou(aram) com o segundo lugar, de acordo com o número de pontos.

Entrada

A entrada contém diversos casos de teste. Jogadores são identificados por inteiros de 1 a 10000. A primeira linha de um caso de teste contém dois inteiros **N** e **M**, indicando, respectivamente, o número de rankings disponíveis ($2 \leq N \leq 500$) e o número de jogadores em cada ranking ($2 \leq M \leq 500$). Cada uma das próximas **N** linhas contém a descrição de um ranking semanal. Cada descrição é composta por uma sequência de **M** inteiros, separados por um espaço em branco, identificando os jogadores que apareceram naquele ranking semanal. Você pode assumir que:

- em cada caso de teste há exatamente um melhor jogador e ao menos um segundo melhor jogador,
- cada ranking semanal consiste de **M** jogadores distintos.

O final da entrada é indicado por **N = M = 0**.

Saída

Para cada caso de teste da entrada seu programa deve produzir uma linha de saída, contendo o identificador do jogador que é o segundo melhor, em número de aparições nos rankings. Se há um empate para segundo lugar, imprima os identificadores de todos os segundo colocados, em ordem crescente. Cada identificador produzido deve ser seguido por um espaço em branco.

Exemplo de Entrada	Exemplo de Saída
4 5 20 33 25 32 99 32 86 99 25 10 20 99 10 33 86 19 33 74 99 32 3 6 2 34 67 36 79 93 100 38 21 76 91 85 32 23 85 31 88 1 0 0	32 33 1 2 21 23 31 32 34 36 38 67 76 79 88 91 93 100