

Desempilhando Caixas

Por Alessandro de Luna Almeida  Brasil

Timelimit: 1

Joãozinho e sua família acabaram de se mudar. Antes da mudança, ele colocou todos os seus livros dentro de várias caixas numeradas. Para facilitar a retirada dos livros, ele fez um inventário, indicando em qual caixa cada livro foi colocado, e o guardou na caixa de número 1.

Chegando no seu novo quarto, ele viu que seus pais guardaram as caixas em várias pilhas, arrumadas em fila, com cada pilha encostada na pilha seguinte.

Joãozinho é um garoto muito sistemático. Por isso, antes de abrir qualquer outra caixa, ele quer recuperar seu inventário. Joãozinho também é um garoto muito desajeitado. Para tirar uma caixa de uma pilha, ele precisa que a caixa esteja no topo da pilha e que ao menos um de seus lados, não importa qual, esteja livre (isto é, não tenham nenhuma caixa adjacente).

Para isso, Joãozinho precisa desempilhar algumas das caixas. Como o quarto dele é bem grande, ele sempre tem espaço para colocar as caixas retiradas em outro lugar, sem mexer nas pilhas que os pais dele montaram.

Para minimizar seu esforço, Joãozinho quer que você escreva um programa que, dadas as posições das caixas nas pilhas, determine quantas caixas Joãozinho precisa desempilhar para recuperar seu inventário.

Entrada

A entrada é composta de vários casos de teste. A primeira linha de cada caso de teste contém dois números inteiros **N** e **P**, indicando, respectivamente, o número de caixas e o número de pilhas ($1 \leq P \leq N \leq 1.000$). As caixas são numeradas seqüencialmente de 1 a **N**.

Cada uma das **P** linhas seguintes descreve uma pilha. Cada linha contém um inteiro Q_i , indicando quantas caixas há na pilha i , seguido de um espaço em branco, seguido de uma lista de Q_i números, que são os identificadores das caixas. Os elementos da lista são separados por um espaço em branco.

Todas as pilhas contêm pelo menos uma caixa, e todas as caixas aparecem exatamente uma vez na entrada. As caixas em cada pilha são listadas em ordem, da base até o topo da pilha. Todas as caixas têm o mesmo formato.

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

A entrada deve ser lida da entrada padrão.

Saída

Para cada caso de teste da entrada, seu programa deve imprimir uma única linha, contendo um único inteiro: o número mínimo de caixas, além da caixa 1, que Joãozinho precisa desempilhar para recuperar o seu inventário.

A saída deve ser escrita na saída padrão.

Exemplo de Entrada	Exemplo de Saída
4 3 1 3 2 1 2	2 0

1 4	Exemplo de Entrada	Exemplo de Saída
4 3		
1 3		
2 2 1		
1 4		
0 0		

Maratona de Programacao da SBC 2007.