

Linhas de Ônibus

Nome do arquivo: `linhas.c`, `linhas.cpp`, `linhas.pas`, `linhas.java`, `linhas.js`, `linhas_py2.py` ou `linhas_py3.py`

Nessa grande cidade na China, há T terminais de ônibus, numerados de 1 a T ; e L linhas de ônibus, numeradas de 1 a L . Os mapas são muito confusos mas conseguimos entender que os ônibus de uma linha fazem viagens circulares passando por um conjunto fixo de terminais. Por exemplo, a tabela seguinte indica o conjunto de terminais por onde passam os ônibus de cada linha, para $T = 10$ e $L = 5$:

Linha	Conjunto de Terminais
1	{4, 3, 8, 2, 1}
2	{5, 10, 7}
3	{1, 5}
4	{6, 8, 10}
5	{9, 4, 5}

Não estamos preocupados com o trajeto da linha, com a ordem na qual o ônibus passa pelos terminais. Portanto, para ir do terminal 2 para o terminal 4, precisamos apenas tomar um ônibus da linha 1 e esperar até ele chegar no terminal 4. O sistema garante que é possível viajar entre qualquer par de terminais, mas talvez seja preciso trocar de linha de ônibus algumas vezes.

Nós estamos com medo de tomar um ônibus errado e acabar perdidos na cidade. É tudo muito grande na China! Por isso, queremos trocar de ônibus o menor número possível de vezes. Por exemplo, você pode ir do terminal 2 para o terminal 10 primeiro tomando a linha 1 até o terminal 1, depois a linha 3 até o terminal 5 e, por fim, a linha 2 até o terminal 10; trocando de ônibus duas vezes, usando três linhas no total. Só que dá para ir do terminal 2 para o 10 trocando apenas uma vez: primeiro tomando a linha 1 até o terminal 8 e depois a linha 4 até o terminal 10.

Neste problema, dados os conjuntos de terminais de cada linha, um terminal origem e um terminal destino, seu programa deve computar o número mínimo possível de linhas de ônibus para fazer a viagem.

Entrada

A primeira linha da entrada contém quatro inteiros, T , L , O e D , representando, respectivamente, o número de terminais, o número de linhas de ônibus, o terminal origem e o terminal destino. As últimas L linhas da entrada descrevem, cada uma, o conjunto de terminais pelos quais uma linha de ônibus passa. A i -ésima linha (dessas últimas L linhas da entrada) descreve o conjunto de terminais da linha de ônibus i , no seguinte formato: o primeiro inteiro na linha, C , indica o número de terminais no conjunto. Depois desse inteiro, o restante da linha da entrada contém C inteiros distintos representando os terminais.

Saída

Seu programa deve produzir uma única linha, contendo apenas um inteiro, o número mínimo possível de linhas de ônibus para viajar do terminal O para o terminal D .

Restrições

- $2 \leq T \leq 500$
- $1 \leq L \leq 500$
- $2 \leq C \leq T$
- $O \neq D$

Informações sobre a pontuação

- Em um conjunto de casos de teste somando 5 pontos, $L = 2$
- Em um conjunto de casos de teste somando outros 5 pontos, $T = 3$
- Em um conjunto de casos de teste somando outros 10 pontos, $T \leq 10$
- Em um conjunto de casos de teste somando outros 20 pontos, $T \leq 100$
- Em um conjunto de casos de teste somando outros 20 pontos, $C \leq 10$
- Em um conjunto de casos de teste somando os demais 40 pontos, nenhuma restrição adicional

Exemplos

Exemplo de entrada 1 10 5 2 10 5 4 3 8 2 1 3 5 10 7 2 1 5 3 6 8 10 3 9 4 5	Exemplo de saída 1 2
Exemplo de entrada 2 2 1 1 2 2 2 1	Exemplo de saída 2 1
Exemplo de entrada 3 10 9 1 10 2 1 2 2 2 3 2 3 4 2 4 5 3 5 6 7 2 6 7 2 7 8 2 8 9 2 9 10	Exemplo de saída 3 8