

## B. Ovenmasters (ovenmasters)

Limite de tempo: 2 segundos

Limite de memória: 1024 MiB

És um repórter no “Excellent Glutenous Ovenmasters of Italy”, um evento onde os melhores  $N$  padeiros de pizza de Itália acabaram de competir para determinar quem faz a melhor pizza. Cada padeiro fez uma pizza, e as pizzas foram depois classificadas por um júri. Cada pizza recebeu uma classificação distinta de 0 (a melhor) a  $N - 1$  (a pior). Cada padeiro recebeu a mesma classificação que a sua pizza.

Após a competição, chegou a hora de comer as pizzas na gala de pizza. Todos os padeiros comparecerão ao evento, e cada um trará a sua própria pizza para a gala. Os padeiros chegam um a um por uma certa ordem (não necessariamente por classificação). Na gala, existem  $M \leq N$  mesas, numeradas de 0 a  $M - 1$ . Os primeiros  $M$  padeiros que chegam colocam as suas pizzas nestas mesas, de 0 a  $M - 1$  pela ordem de chegada. Cada um dos restantes  $N - M$  padeiros gostaria de comer uma pizza melhor que a sua, mas não demasiado boa, para não se sentirem mal consigo próprios. Cada vez que um padeiro chega, escolhe a pizza disponível com a pior classificação que ainda seja melhor que a sua. Sentam-se na mesa correspondente para comer a pizza escolhida toda. Finalmente, deixam a sua própria pizza para trás na mesma mesa para que outro padeiro a possa potencialmente comer mais tarde. Se não existir uma pizza adequada para um padeiro que chega (porque todas as mesas têm pizzas com classificação pior que a sua), o padeiro vai-se embora frustrado e leva a sua pizza consigo.

O exemplo seguinte mostra uma gala com  $M = 2$  mesas e padeiros a chegar na seguinte sequência de classificações: 1, 0, 3, 5, 4, 2. Esta gala corresponde ao primeiro exemplo de input e output.

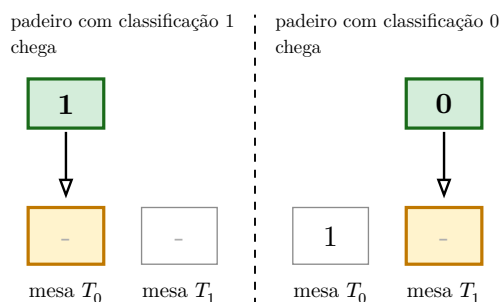


Figura 1: Os primeiros  $M = 2$  padeiros colocam as suas pizzas nas mesas vazias ( $T_0, T_1$ ) por ordem de chegada.

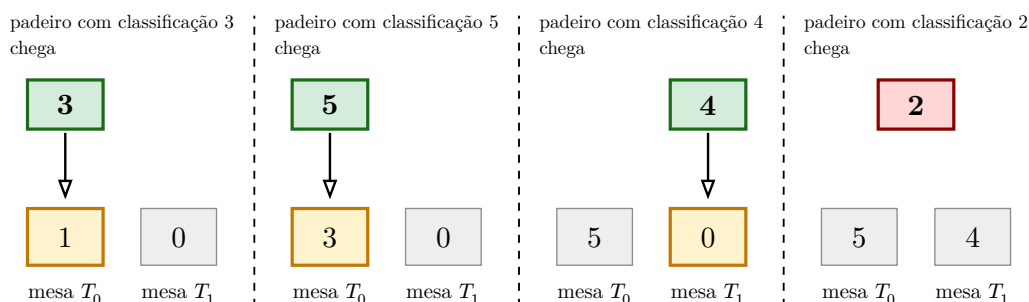


Figura 2: Quando todas as mesas estiverem ocupadas, cada padeiro que chega vai à mesa com a pizza pior que ainda é melhor que a sua (indicado pela seta), come essa pizza, e deixa a sua. Se não existir uma pizza melhor, o padeiro vai-se embora frustrado (sem seta).

No teu artigo, queres relatar a ordem pela qual os padeiros chegaram à gala de pizza. Infelizmente, estavas demasiado distraído com todas as pizzas saborosas e esqueceste-te de apontar a ordem pela qual os padeiros chegaram. Felizmente, em cada mesa, podes encontrar uma pilha de tabuleiros das pizzas que foram servidas nessa mesa, por ordem em que as pizzas foram servidas.

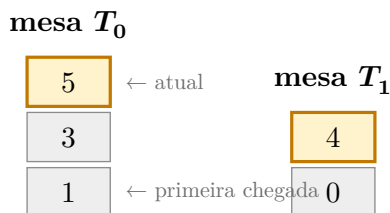


Figura 3: Pilhas de tabuleiros correspondentes ao primeiro exemplo. Cada pilha lista os padeiros que estiveram nessa mesa por ordem de chegada, do fundo (primeiro) para o topo (mais recente). O tabuleiro d tem a pizza que foi deixada lá no fim da gala.

Queres usar esta informação para reconstruir a ordem pela qual os padeiros chegaram. Estás ciente de que podem ter existido várias ordens possíveis, por isso, para a pontuação total, queres relatar a ordem válida lexicograficamente menor.<sup>1</sup>

## Input

A primeira linha contém dois números inteiros  $N$  e  $M$ , o número de padeiros e o número de mesas.

Seguem-se  $M$  linhas, cada uma descrevendo uma pilha de tabuleiros numa mesa. A linha  $i$  começa com um número inteiro  $T_i$ , o número de tabuleiros na mesa  $i$ , seguido por  $T_i$  números inteiros  $b_{i,j}$  indicando a classificação da  $j$ -ésima pizza que foi servida na mesa  $i$ .

## Output

Escreve **NO** se não houver uma ordem possível que satisfaça as restrições. Escreve **YES** se houver uma ordem possível. Nesse caso, escreve uma segunda linha contendo  $N$  inteiros  $a_0, a_1, \dots, a_{N-1}$ , as classificações dos padeiros por ordem de chegada. Se existirem várias permutações desse tipo, deves escrever a lexicograficamente menor. Nota que respostas parcialmente corretas podem ainda pontuar, como explicado na secção de Pontuação.

## Restrições

- $1 \leq M \leq N \leq 300\,000$ .
- $0 \leq b_{i,j} \leq N - 1$ .
- Todos os  $b_{i,j}$  são distintos.
- $1 \leq T_i \leq N$ .

## Pontuação

O teu programa será testado em vários casos de teste agrupados em subtarefas. Para obter a pontuação de uma subtarefa, tens de resolver corretamente todos os testes que ela contém.

Soluções apenas com uma primeira linha correta (**YES** vs **NO**) pontuarão 20%. Soluções com uma primeira linha correta (**YES** vs **NO**) e **qualquer ordem válida**, não necessariamente a lexicograficamente menor, quando a resposta é **YES** pontuarão mais 20%. Para pontuar os restantes 60%, tens de escrever a ordem válida lexicograficamente menor quando a primeira linha é **YES**.

- **Subtarefa 0 [ 0 pontos]**: Exemplos.

<sup>1</sup>Uma sequência  $a_0, a_1, \dots, a_{n-1}$  é lexicograficamente menor que uma sequência  $b_0, b_1, \dots, b_{n-1}$  se existir um índice  $0 \leq t < n$  tal que  $a_i = b_i$  para todo o  $i < t$  e  $a_t < b_t$ .

- **Subtarefa 1 [20 pontos]:**  $M = 1$ .
- **Subtarefa 2 [10 pontos]:**  $M = 2$ ,  $N \leq 200$ , e a soma de todos os  $T_i$  é  $N$  (por outras palavras, nenhum padeiro vai-se embora frustrado).
- **Subtarefa 3 [20 pontos]:**  $M \leq N \leq 200$ , e a soma de todos os  $T_i$  é  $N$  (por outras palavras, nenhum padeiro vai-se embora frustrado).
- **Subtarefa 4 [20 pontos]:**  $M \leq 10$ .
- **Subtarefa 5 [30 pontos]:** Sem restrições adicionais.

## Exemplos

stdin	stdout
6 2 3 1 3 5 2 0 4	YES 1 0 3 5 4 2
6 2 3 1 3 4 2 0 2	NO
4 2 2 0 3 2 1 2	NO
3 1 2 0 2	YES 0 2 1
8 1 8 7 6 5 4 3 2 1 0	NO
12 4 3 2 3 4 1 5 1 6 5 7 8 9 10 11	YES 2 5 6 7 0 1 3 4 8 9 10 11

## Explicação

O primeiro exemplo de input e output corresponde às figuras mostradas no enunciado do problema. Em particular, a ordem pela qual os padeiros chegam à gala nas Figuras 1 e 2 é a ordem de chegada válida lexicograficamente menor 1, 0, 3, 5, 4, 2.

No segundo exemplo, as pilhas de tabuleiros são inconsistentes já que não existe nenhuma ordem de chegada na qual o padeiro 5 sairia frustrado. Logo a resposta é NO.

No terceiro e quinto exemplos, as pilhas de tabuleiros também são inconsistentes (nenhuma ordem de chegada as pode produzir), por isso a resposta é NO.

No quarto exemplo ( $N = 3$ ,  $M = 1$ ) apenas uma ordem de chegada é possível, nomeadamente 0, 2, 1.

No sexto exemplo ( $N = 12$ ,  $M = 4$ ) nota que os números 0 e 1 não aparecem entre os valores  $b_{i,j}$ . Isto significa que em algum momento durante a gala cada um dos padeiros 0 e 1 foi embora frustrado. O output do exemplo mostra a ordem de chegada válida lexicograficamente menor. Existem outras ordens de chegada válidas; por exemplo 2, 5, 6, 7, 8, 1, 3, 4, 9, 10, 11, 0. Escrever YES seguido por uma ordem válida alternativa como esta (em vez da lexicograficamente menor) seria considerado parcialmente correto para 40% da pontuação.