

B. 披薩大師 (ovenmasters)

時間限制: 2 秒

空間限制: 1024 MiB

你是一位在「意大利絕佳筋道披薩大師賽」(Excellent Glutenous Ovenmasters of Italy) 的記者，這場剛結束的比賽中，意大利最強的 N 位披薩師傅決出了誰的披薩做得最好。每位師傅各烤了一份披薩，評審團對這些披薩進行了排名。每份披薩都獲得了一個 0 (最好) 到 $N - 1$ (最差) 的獨特名次。每位師傅獲得的名次與其披薩相同。

比賽結束後，來到披薩晚宴享用披薩的時間。所有師傅都來赴宴，且每個人都會帶上自己的披薩去晚宴。師傅們會依照某種順序（不一定按名次）逐一到場。晚宴現場有 $M \leq N$ 張桌子，編號從 0 到 $M - 1$ 。前 M 位抵達的師傅會按抵達順序將他的披薩放在桌子上，編號分別為 0 到 $M - 1$ 。剩下的 $N - M$ 位師傅每個人都想吃一份比自己做的更好，但又不會太好（以免讓他們感到自卑）的披薩。每當有師傅抵達，他會選擇還在桌上的、名次優於自己的披薩中，名次最差的一個。他會在對應的桌子坐下，吃掉整份選好的披薩。最後，他會把自己的披薩留在桌上，供之後的其他師傅可能取用。如果對新到的師傅來說沒有合適的披薩（因為所有桌上的披薩名次都比他自己的差），該師傅會感到沮喪並帶著自己的披薩離開。

下面的例子展示了一個有 $M = 2$ 張桌子的晚宴，師傅抵達的順序為：1, 0, 3, 5, 4, 2。這對應了第一個測試輸入與輸出。

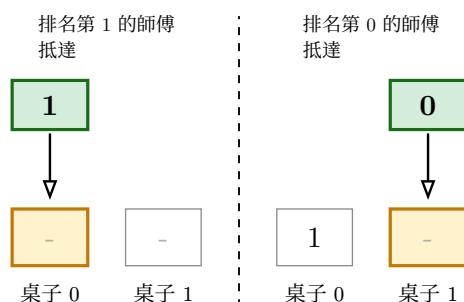


图 1 前 $M = 2$ 位師傅依抵達順序將披薩放在空桌 (0, 1) 上。

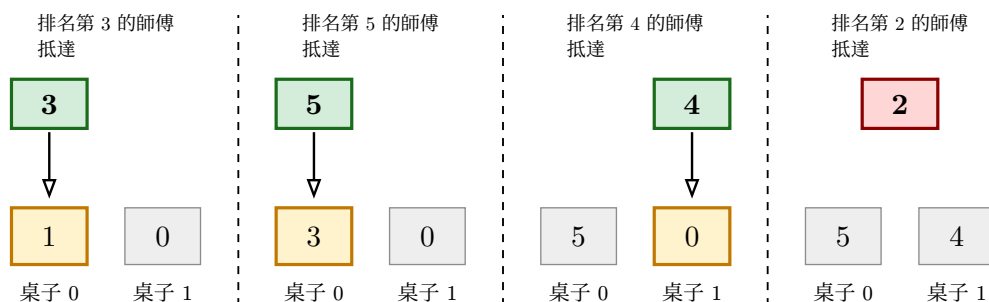


图 2 一旦所有桌子都被佔用，每位新到的師傅會走向那張放著「比自己做的披薩排名要好、但卻是當中排名最差」的披薩的桌子（箭頭所示），吃掉那份披薩，然後留下自己的。如果不存在這樣的披薩，師傅就會沮喪地離開（沒有箭頭）。

在你的報導中，你本來想記錄師傅們抵達披薩晚宴的順序。不幸的是，你被美味的披薩分散了注意力，忘了記下他們抵達的順序。幸運的是，在每一張桌子上，你可以找到端上該桌的披薩的一疊餐盤，順序是它們端上桌的順序。

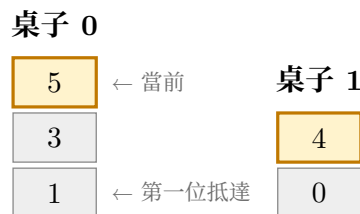


图 3 對應第一個例子的餐盤堆。每疊餐盤列出了曾在該桌的師傅，按抵達順序從下（第一位）到上（最近期）。高亮的餐盤是晚宴結束時留在該桌的披薩。

你想利用這些信息還原師傅們抵達的順序。你知道可能有多種可能的順序，所以為了獲得滿分，你需要報告字典序最小的合規順序。¹

輸入

第一行包含兩個整數 N 和 M ，分別為師傅的人數和桌子的數量。

接著有 M 行，每一行描述一張桌子上的餐盤堆。第 i 行以一個整數 T_i 開頭，代表桌子 i 上的餐盤數量，隨後是 T_i 個整數 $b_{i,j}$ ，表示在桌子 i 上享用的第 j 份披薩的排名。

輸出

如果不存在滿足條件的順序，輸出 NO。如果存在可能的順序，輸出 YES。在這種情況下，輸出第二行，包含 N 個整數 a_0, a_1, \dots, a_{N-1} ，即按抵達順序的師傅排名。如果存在多種這樣的排列，你應該輸出字典序最小的那一個。注意，部分正確的答案仍可能獲得一些分數，詳見「評分方式」一節。

限制條件

- $1 \leq M \leq N \leq 300\,000$.
- $0 \leq b_{i,j} \leq N - 1$.
- 所有 $b_{i,j}$ 均互不相同。
- $1 \leq T_i \leq N$.

評分方式

你的程式將以多筆測資進行評測，測資分成若干個子任務。要獲得一個子任務的分數，你必須正確解決其中包含的所有測試資料。

只有第一行 (YES 或 NO) 正確的解法可得 20% 分數。如果第一行正確且在答案為 YES 時輸出 **任何合規** 的順序 (即使不是字典序最小)，可另外獲得 20% 分數。要獲得剩下的 60%，你必須在答案為 YES 時輸出字典序最小的有效順序。

- 子任務 0 [0 分]: 範例。
- 子任務 1 [20 分]: $M = 1$.
- 子任務 2 [10 分]: $M = 2$, $N \leq 200$ ，且所有 T_i 的總和為 N (換句話說，沒有師傅會沮喪地離開)。
- 子任務 3 [20 分]: $M \leq N \leq 200$ ，且所有 T_i 的總和為 N (換句話說，沒有師傅會沮喪地離開)。
- 子任務 4 [20 分]: $M \leq 10$.
- 子任務 5 [30 分]: 無額外限制。

¹序列 a_0, a_1, \dots, a_{n-1} 在字典序上小於序列 b_0, b_1, \dots, b_{n-1} ，意思是存在一個下標 $0 \leq t < n$ ，使得對於所有 $i < t$ ，都有 $a_i = b_i$ ，且 $a_t < b_t$ 。

範例

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

說明

第一個範例的輸入和輸出對應問題敘述的附圖。特別是，圖 1 和圖 2 中師傅抵達晚宴的順序是字典序最小的合規抵達順序 1, 0, 3, 5, 4, 2。

在第二個範例中，餐盤堆不一致，因為沒有任何抵達順序能讓排名第 5 的師傅沮喪離開。因此答案是 NO。

在第三和第五個範例中，餐盤堆也是不一致的（沒有抵達順序能產生這些堆），所以答案是 NO。

在第四個範例 ($N = 3, M = 1$) 中，只有一種抵達順序是可能的，即 0, 2, 1。

在第六個範例 ($N = 12, M = 4$) 中，請注意數字 0 和 1 沒有出現在 $b_{i,j}$ 的值中。這意味著在晚宴期間，師傅 0 和 1 都曾沮喪地離開。範例輸出顯示了字典序最小的合規抵達順序。也有其他合規的抵達順序；例如 2, 5, 6, 7, 8, 1, 3, 4, 9, 10, 11, 0。輸出 YES 並加上像這樣的替代合規順序（而不是字典序最小的那個）會被視為部分正確，獲得 40% 的分數。