

## B. Ovenmasters (ovenmasters)

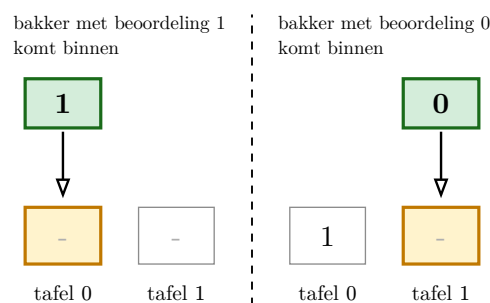
Tijdslimiet: 2 seconden

Geheugenlimiet: 1024 MiB

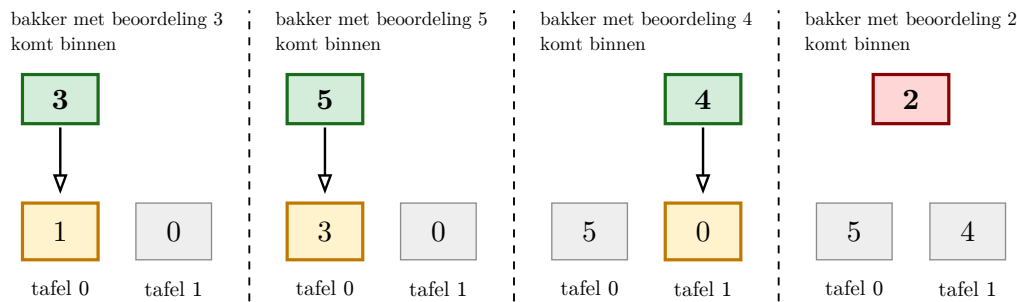
Je bent een verslaggever bij de “Excellente Glutenvolle Ovenmasters van Italië”, een evenement waar de beste  $N$  pizzabakkers van Italië net gestreden hebben om te bepalen wie de beste pizza bakt. Elke bakker heeft één pizza gebakken en de pizza’s werden daarna door een jury beoordeeld. Elke pizza kreeg daarbij een unieke beoordeling van 0 (beste) tot en met  $N - 1$  (slechtste). Elke bakker kreeg vervolgens dezelfde beoordeling als haar pizza.

Na de wedstrijd is het tijd om de pizza’s op te eten tijdens het pizzagala. Alle bakkers zijn aanwezig bij het evenement en ze nemen allemaal hun eigen pizza mee naar het gala. De bakkers komen één voor één aan in één of andere volgorde (niet noodzakelijkerwijs op volgorde van beoordeling). Op het gala zijn  $M \leq N$  tafels, genummerd van 0 tot en met  $M - 1$ . De eerste  $M$  bakkers die aankomen, plaatsen hun pizza’s op tafel 0 tot en met  $M - 1$  in volgorde van aankomst. Elk van de overige  $N - M$  bakkers wil graag een pizza eten die beter is dan die van haarzelf, maar niet té goed, zodat ze zich niet slecht voelt over zichzelf. Elke keer dat een bakker aankomt, kiest ze van de beschikbare pizza’s die met de slechtste beoordeling die nog wel beter is dan die van haarzelf. Ze gaat aan de bijbehorende tafel zitten om de gekozen pizza helemaal op te eten. Tenslotte laat ze haar eigen pizza achter op dezelfde tafel zodat een andere bakker die eventueel later kan opeten. Als er geen geschikte pizza is voor een binnenkomende bakker (omdat op alle tafels pizza’s liggen die slechter zijn dan die van haarzelf), vertrekt de bakker gefrustreerd en neemt haar pizza mee.

Het volgende voorbeeld laat een gala zien met  $M = 2$  tafels en bakkers die aankomen in de volgende volgorde van beoordeling: 1, 0, 3, 5, 4, 2. Dit gala komt overeen met de eerste voorbeeldinvoer en -uitvoer.

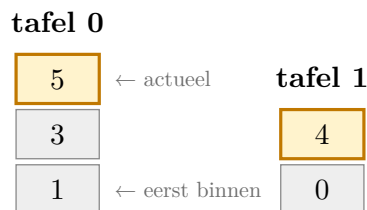


Figuur 1: De eerste  $M = 2$  bakkers plaatsen hun pizza’s op de lege tafels (0, 1) in volgorde van binnenkomst.



Figuur 2: Zodra alle tafels bezet zijn, gaat elke binnenkomende bakker naar de tafel met de slechtste pizza die nog wel beter is dan die van haarzelf (aangegeven door de pijl), eet die pizza op en laat haar eigen pizza achter. Als er geen betere pizza bestaat, vertrekt de bakker gefrustreerd (geen pijl).

In je artikel wil je verslag doen van de volgorde waarin de bakkers op het pizzagala binnenkwamen. Helaas was je te veel afgeleid door alle lekkere pizza's en vergat je de volgorde te op te schrijven waarin de bakkers binnenkwamen. Gelukkig zie je op elke tafel een stapel dienbladen waarop de pizza's aan deze tafel werden geserveerd, in de volgorde waarin de pizza's werden achtergelaten.



Figuur 3: Stapels dienbladen die horen bij het eerste voorbeeld. Elke stapel geeft de bakkers die aan die tafel zaten in volgorde van binnenkomst, van onder (eerst) naar boven (meest recent). Het gemarkeerde dienblad bevat de pizza die aan het einde van het gala op die plek lag.

Je wilt deze informatie gebruiken om de volgorde waarin de bakkers binnenkwamen te reconstrueren. Je bent je ervan bewust dat er verschillende mogelijke volgordes kunnen zijn, dus voor de volledige score wil je die geldige volgorde rapporteren die in woordenboekvolgorde het kleinst mogelijk is.<sup>1</sup>

## Invoer

De eerste regel bevat twee gehele getallen  $N$  en  $M$ , het aantal bakkers en het aantal tafels.

Daarna volgen  $M$  regels. Elke regel beschrijft een stapel dienbladen op een tafel. De  $i$ -de regel begint met een geheel getal  $T_i$ , het aantal dienbladen op tafel  $i$ . Daarna komen  $T_i$  gehele getallen  $b_{i,j}$  die de beoordeling aangeven van de  $j$ -de pizza die op tafel  $i$  werd achtergelaten.

## Uitvoer

Voer **NO** uit als er geen mogelijke volgorde is die aan de eisen voldoet. Voer **YES** uit als er een mogelijke volgorde is. Voer in dat geval op de tweede regel  $N$  gehele getallen  $a_0, a_1, \dots, a_{N-1}$  uit, de beoordelingen van de bakkers in volgorde waarop ze binnenkwamen. Als er meerdere oplossingen bestaan, moet je de in woordenboekvolgorde kleinste daarvan uitvoeren. Let op dat gedeeltelijk correcte antwoorden nog steeds punten kunnen opleveren. Dit wordt uitgelegd in de sectie 'Scoring'.

## Randvoorwaarden

- $1 \leq M \leq N \leq 300\,000$ .
- $0 \leq b_{i,j} \leq N - 1$ .
- Alle  $b_{i,j}$  zijn verschillend.
- $1 \leq T_i \leq N$ .

<sup>1</sup>Een rij  $a_0, a_1, \dots, a_{n-1}$  is in woordenboekvolgorde kleiner dan een rij  $b_0, b_1, \dots, b_{n-1}$  als er een index  $0 \leq t < n$  bestaat zodat geldt dat  $a_i = b_i$  voor alle  $i < t$  en  $a_t < b_t$ .

## Scoring

Je programma wordt getest op verschillende testgevallen. Deze zijn gegroepeerd in deelopgaven (subtasks). Om de punten voor een subtaak te behalen, moet je alle tests van deze subtask correct oplossen.

Oplossingen waarvan alleen de eerste regel correct is (YES vs NO) scoren 20%. Oplossingen met een correcte eerste regel (YES vs NO) die bij het antwoord 'Yes' een **willekeurige** geldige volgorde geven die niet noodzakelijkerwijs de in woordenboekvolgorde kleinste is, scoren nog eens 20%. Om de resterende 60% te scoren, moet je de in woordenboekvolgorde kleinste geldige volgorde uitvoeren als de eerste regel YES is.

- **Subtask 0 [ 0 punten]:** Voorbeelden.
- **Subtask 1 [20 punten]:**  $M = 1$ .
- **Subtask 2 [10 punten]:**  $M = 2$ ,  $N \leq 200$ , en de som van alle  $T_i$  is  $N$  (met andere woorden, geen enkele bakker loopt gefrustreerd weg).
- **Subtask 3 [20 punten]:**  $M \leq N \leq 200$ , en de som van alle  $T_i$  is  $N$  (met andere woorden, geen enkele bakker loopt gefrustreerd weg).
- **Subtask 4 [20 punten]:**  $M \leq 10$ .
- **Subtask 5 [30 punten]:** Geen extra randvoorwaarden.

## Voorbeelden

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

## Uitleg

De invoer en uitvoer van het eerste voorbeeld horen bij de figuren in de tekst van de opgave. De in woordenboekvolgorde kleinste geldige volgorde waarin de bakkers op het gala aankomen in Figuur 1 en 2 is 1, 0, 3, 5, 4, 2.

In het tweede voorbeeld zijn de stapels dienbladen inconsistent, aangezien er geen aankomstvolgorde is waarin de bakker met beoordeling 5 gefrustreerd zou vertrekken. Daarom is het antwoord NO.

In het derde en vijfde voorbeeld zijn de stapels dienbladen ook inconsistent (geen enkele aankomstvolgorde kan deze stapels produceren), dus het antwoord is NO.

In het vierde voorbeeld ( $N = 3$ ,  $M = 1$ ) is er maar één aankomstvolgorde mogelijk, namelijk 0, 2, 1.

In het zesde voorbeeld ( $N = 12$ ,  $M = 4$ ), kun je zien dat de getallen 0 en 1 niet voorkomen tussen de waarden  $b_{i,j}$ . Dit betekent dat de bakkers 0 en 1 allebei op een gegeven moment tijdens het gala gefrustreerd wegliepen. De voorbeelduitvoer laat de in woordenboekvolgorde kleinste geldige aankomstvolgorde zien. Er bestaan andere geldige aankomstvolgordes; bijvoorbeeld 2, 5, 6, 7, 8, 1, 3, 4, 9, 10, 11, 0. Het uitvoeren van YES gevolgd door een alternatieve geldige volgorde zoals deze (in plaats van de in woordenboekvolgorde kleinste) zou als gedeeltelijk correct worden beoordeeld met 40% van de score.