

CALLSCHE: Call Center Schedule

Легенда, описанная далее, переформулирована и упрощена переводчиком, чтобы читатель мог лучше понять условие задачи. Оригинальную легенду вы можете прочитать на странице задачи в контексте.

Условие:

Шеф работает менеджером в call-центре. У него есть много подчиненных. Каждый из них проводит свое время участвуя в деловых встречах, разговаривая по телефону с клиентами или просто занимается какими-то другими делами по офису.

Каждый час работы каждого сотрудника должен быть выделен на какой-то один тип работы(деловая встреча, разговор по телефону или другие дела в офисе). Сотрудник не может посреди часа менять тип работы.

Пусть у нас есть D рабочих дней в неделе, пронумерованных числами от 1 до D . i -й сотрудник может разговаривать по телефону не больше, чем L_i рабочих часов в неделю. Для каждого сотрудника наперед известны рабочие часы отведенные на деловые встречи.

Call-центр работает ровно N часов каждый день(пронумеруем их целыми числами от 1 до N).

Для каждого часа недели известно сколько человек позвонит в call-центр. Другими словами, Chef знает, что ему нужно ровно $R_{i,j}$ сотрудников, которые будут говорить по телефону с клиентами в j -й час i -го дня недели.

$F_{k,i,j}$ равно 1 если k -й сотрудник свободен для разговора по телефону с клиентами в j -й час i -го дня, и 0 если у него на этот час назначена деловая встреча.

Заметьте, что Шеф живет в очень странном мире, где длина дня не обязательно равна 24 часам, а недели — 7 дням.

Шефу нужно создать расписание работы для каждого сотрудника. Не нужно забывать, что еще есть время на обед каждый день с LT_{begin} по LT_{end} часы работы(обе границы включительно). Поэтому Шефу еще нужно следить за тем, чтобы у каждого сотрудника был хотя бы один свободный час для обеда(свободный час в диапазоне $[LT_{begin} \dots LT_{end}]$).

Если у какого-то сотрудника нету назначенной деловой встречи на какой-то час и он не говорит по телефону с клиентом в этот час, он может просто выполнять другую работу по офису в это время и это тоже будет считаться как рабочий час. Также он может провести этот час за работой над каким-то своим личным делом и тогда этот час не будет учитываться как рабочий.

Помогите Шефу узнать, можно ли создать расписание так, чтобы выполнялись следующие условия:

- Каждый сотрудник будет проводить не больше, чем N часов в день за разговорами по телефону и на деловых встречах.

CODECHEF

- Каждый сотрудник будет проводить не больше, чем L_i часов в неделю разговаривая по телефону.
- У каждого сотрудника будет хотя бы один, свободный от разговоров по телефону и от деловых встреч, час в обедний период.
- В j -й час i -го дня будет ровно R_{ij} сотрудников, которые разговаривают по телефону с клиентами.

Формат ввода:

Первая строка содержит ровно одно целое число T — количество тестовых случаев. Описание формата ввода каждого теста следует дальше.

Первая строка каждого тестового случая содержит четыре разделенных пробелами числа — P , D , H и N — обозначающие количество подчиненных Шефа, количество рабочих дней в неделе, количество рабочих часов call-центра в день и количество рабочих часов в день для сотрудников соответственно.

Следующая строка содержит P разделенных пробелами чисел — L_1, L_2, \dots, L_P .

Следующая строка содержит два разделенных пробелами числа — LT_{begin} и LT_{end} — обозначающие час, с которого начинается время для обеда и час, после которого оно заканчивается.

Следующие D строк содержат по H разделенных пробелами чисел. j -е число i -й строки обозначает R_{ij} .

Следующие P блоков содержат по D строк каждый. В каждой строке записано ровно H чисел. j -е число i -й строки k -го блока обозначает F_{kij} .

Формат вывода:

Выведите «Yes» (без кавычек) если возможно составить корректное расписание для сотрудников и «No» (без кавычек) иначе.

Ограничения:

- $1 \leq T \leq 5$
- $1 \leq N \leq H \leq \text{MAX}$
- $1 \leq D, P \leq \text{MAX}$
- $1 \leq L_i \leq N * D$
- $1 \leq R_{ij} \leq 15$
- $0 \leq F_{kij} \leq 1$
- $1 \leq LT_{begin}, LT_{end} \leq N$
- Подзадача 1(15 баллов): $\text{MAX} = 3$
- Подзадача 2(25 баллов): $\text{MAX} = 10$
- Подзадача 3(60 баллов): $\text{MAX} = 70$

Примеры тестов:

Входные данные:

2
2 2 3 2
4 1
2 3
0 1 1
0 1 0
1 1 1
1 1 1
1 1 1
1 0 1
2 2 3 2
4 1
2 3
0 1 2
0 1 0
1 1 1
1 1 1
1 1 1
1 0 1

Выходные данные:

Yes
No