

Multitarea

Los calendarios o agendas electrónicas controlan nuestra vida diaria. Para personas como yo, que no se nos da bien la multitarea, es importante tener como mucho una tarea planificada para cualquier minuto de nuestra vida.



En mi calendario hay dos tipos de tareas: tareas únicas y tareas periódicas. Las tareas únicas tienen un instante de comienzo, c y otro de finalización, f , lo que significa que la tarea me tendrá ocupado durante el intervalo de tiempo $[c, f)$. Las tareas periódicas tienen además de los instantes de comienzo y finalización de su primera aparición, un periodo de repetición. Se supone que estas tareas se repiten para siempre sin fin. Para simplificar las cosas, todos los tiempos se expresan en minutos desde un tiempo inicial 0. Por ejemplo, una tarea periódica con tiempo de inicio 5, tiempo de finalización 8 y periodo de repetición 100 ocurriría en los intervalos de tiempo $[5..8)$, $[105..108)$, $[205..208)$, etc...

Tu trabajo consiste en averiguar si mi calendario está libre de conflictos durante un periodo de mi vida. Se considera que dos tareas están en conflicto si y solo si sus intervalos de tiempo se superponen, por ejemplo $[2..5)$ y $[4..6)$ se superponen, pero no $[2..8)$ con $[9..10)$, o $[2..8)$ con $[8..10)$.

Entrada

La entrada está formada por una serie de casos de prueba, ocupando cada uno de ellos varias líneas.

En la primera línea aparecen tres números: N , que representa el número de tareas únicas; M , que representa el número de tareas periódicas; y T , que representa el número de minutos en los que quiero averiguar si hay o no conflictos. A continuación, aparecen N líneas, cada una con los instantes de comienzo y finalización de una tarea única. A estas les siguen M líneas más, cada una con tres números: el instante de comienzo y finalización de la primera aparición de una tarea repetitiva, y el periodo de repetición.

Siempre hay al menos 1 tarea y nunca más de 10.000. Todos los tiempos son números entre 0 y 10^8 . Se garantiza que todas las tareas tardan al menos un minuto y que los intervalos de una misma tarea periódica no solapan entre sí.

Salida

Para cada caso de prueba el programa escribirá SI si hay conflictos entre algunas tareas en el periodo de tiempo $[0..T)$, y NO en caso contrario.

Entrada de ejemplo

```
2 0 10
2 5
4 6
0 2 100
```

1 4 8
5 7 8
2 1 10
8 20
1 5
6 7 10

Salida de ejemplo

SI
NO
NO