

Organizando los hangares

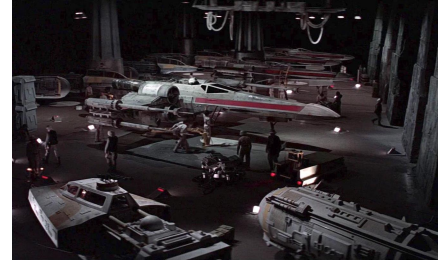
Tiempo máximo: 2,000-3,000 s Memoria máxima: 4096 KiB

<http://www.aceptaelreto.com/problem/statement.php?id=429>

Las Fuerzas Armadas de la Alianza están teniendo serios problemas logísticos en los hangares de las naves. Durante las batallas y escaramuzas con el enemigo, muchas tienen que ponerse a resguardo en bases diferentes a las de partida, generando bastante desorden en la gestión del espacio.

Cada base posee varios hangares de tamaños variados. Los responsables de enviar las naves a los hangares utilizan como única directriz asignar a cada nave el hangar que más hueco tenga disponible en ese momento. Gracias a complejos estándares de tamaños y formas, tienen garantizado que si hay espacio suficiente, la nave entrará.

Por desgracia, hay veces que llega una nave grande para la que no hay espacio disponible suficiente. Muchas veces la nave habría podido entrar si las anteriores se hubieran enviado a hangares distintos, pero ya es demasiado tarde y no hay forma de admitirla. Para la preparación de batallas futuras, se quiere hacer algunas simulaciones de llegadas de naves a bases para detectar con qué frecuencia ocurre esta situación, y buscar estrategias diferentes.



Entrada

El programa deberá leer, de la entrada estándar, múltiples casos de prueba. Cada uno comienza con un número $1 \leq H \leq 10$ indicando la cantidad de hangares disponibles en una determinada base, seguido de H números con sus tamaños.

A continuación, aparece el número $0 \leq N \leq 200$ con la cantidad de naves que llegan a la base, seguido de N números con sus tamaños, por orden de llegada. Todos los hangares están vacíos cuando llega la primera nave.

El tamaño de cada hangar y de cada nave es mayor que 0 y menor o igual que 1.000.000.

La entrada termina con un 0.

Salida

Para cada caso de prueba el programa escribirá "SI" si todas las naves pueden guardarse en los hangares siguiendo la estrategia de asignación descrita, y "NO" en caso contrario.

Entrada de ejemplo

```
2
12 20
3
10 11 10
2
12 20
3
10 10 11
3
20 5 10
5
14 3 4 6 5
0
```

Salida de ejemplo

SI
NO
SI

Autores: Alfredo Oltra Orengo y Sergi García Barea.

Revisores: Pedro Pablo Gómez Martín y Marco Antonio Gómez Martín.