

Problema A

Pánico en el Supermercado

Con el nuevo brote de Culonavirus, el gobierno ha declarado una cuarentena general. En pánico por lo que acaba de pasar, la gente (Tú incluido) se ha armado de valor, con guantes y mascarilla, y ha salido a comprar al supermercado, para coger provisiones de comida para los próximos días. Ya en el supermercado, al abrir las puertas has empezado a correr con tu carrito y has echado lo primero que has encontrado en las baldas. No has mirado ni qué era, ni si te gustaba o no, ni como venía. Tu has comprado lo que has podido y directo a casa. El problema lo has encontrado cuando, al llegar a casa, has visto la despensa. Tienes apenas un par de armarios y de baldas, y has hecho espacio como has podido. Ahora te preguntas si podrás meter toda la compra en la despensa. Por suerte, algunos productos vienen en packs, así que esos productos has decidido dividirlos de forma individual y meterlos en la despensa divididos. A fin de cuentas, no te importa compartir una lata de maiz con la vecina con tal de poder colocar las cosas más fácilmente. Dada la lista de productos que has comprado, lo que ocupa cada uno, y si son o no divisibles, junto a la lista de huecos en la despensa, ¿Podrías decirnos si cabe la comida en la despensa, o si por el contrario, tienes que regalar comida a tus vecinos?

Input

La entrada estará compuesta por un primer número C , indicando el número de casos de prueba, cada uno comenzando por un número N de productos comprados. A continuación, acompañarán N líneas, cada una con dos números, T y D . El primero, T , es el tamaño de un producto completo, y a continuación D , el número de veces que se puede dividir ese producto. ($D = 1$ significa que el producto es indivisible). Se garantiza que todas las divisiones de productos serán exactas. Después aparecerá un número H indicando el número de huecos en la despensa, seguido de H líneas, cada una con el tamaño de cada hueco en la despensa. Solo se puede colocar un producto (O una división de un producto) en cada hueco de la despensa.

La entrada debe ser leída de forma estándar.

Output

Para cada caso de prueba, la salida será una sola línea indicando “SUFICIENTE HUECO” si cabe todo en la despensa al colocar los productos, o “REGALAR COMIDA A LA VECINA” si los productos no caben en la despensa.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
2 5 6 1 7 1 8 4 9 3 2 1 10 2 2 2 6 3 3 2 3 7 2 2 16 2 4 1 2 10 20	SUFICIENTE HUECO REGALAR COMIDA A LA VECINA

Constraints

- $1 \leq N \leq 10000$
- $1 \leq D_i \leq T_i \leq 10000$
- $0 \leq H \leq 300000$
- $1 \leq H_i \leq 10000$

Problema B

El parque de atracciones LongLongLandia

Los gerentes del parque de atracciones de LongLongLandia necesitan estimar cual es la altura mínima para que pasen gratis sus futuros clientes (de esta manera podrán aumentar sus ganancias), por eso mismo estan realizando estimaciones sobre las alturas con las atracciones más relevantes. Para ser exactos en las alturas, los gerentes que son muy peculiares comienzan a medir a la gente en nanometros, de esta forma no se equivocarán en la precisión.

Input

En la entrada se garantiza que dos personas no tendrán la misma altura debido a la precisión que se tiene de la medida de la altura. El primer número T se corresponde con el número de casos de prueba, posteriormente viene un número F de filas que tiene la atracción. Para cada fila se tiene un número P de personas que contiene la fila, en la siguiente línea se tendrá una lista de las P personas que vienen a continuación, donde cada persona tendrá H altura. Finalmente tendremos un número Q para las diferentes consultas que desean realizar los gerentes y dichas Q consultas.

La entrada debe ser leída de forma estándar.

Output

Para cada caso de prueba, los gerentes del parque de atracciones te piden que para cada consulta devuelvas cuantas personas son más bajas que el número que te dan.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
1	2
2	5
3	2
180000000 190000000 195000000	
2	
185000000 191000000	
3	
185000001 196000000 190000000	

Constraints

- $1 \leq T \leq 100000$
- $1 \leq F \leq 1000$
- $1 \leq P \leq 300$
- $1 \leq H \leq 10^{18}$
- $1 \leq Q \leq 150000$

Problema C

Envios Prioritarios

Amazon está pensando añadir un nuevo tipo de suscripción a su servicio de compras online. El nuevo servicio lleva las siglas de “P2W”, nadie está seguro del significado que llevan detrás. Los usuarios que opten por pagar los 100 euros que cuesta podrán hacer que sus envíos se salten posiciones en la cola de salida de los centros de reparto; en concreto, saldrán antes que cualquier producto que no se haya pedido con este servicio.

Entrada

La entrada contendrá un único caso de prueba.

El caso empezará con un número, N , que indicará el número de eventos que sucederán a lo largo de una semana en el centro de reparto. Cada evento ocupará una línea y puede tener uno de los siguientes formatos:

- “P2W” o “NOR” - indicando la llegada de un paquete “P2W” o normal a la cola de reparto.
- “R C ” - indicando que va a salir un camión con capacidad para C paquetes del centro de reparto.

Tanto en la cola de reparto como en el camión los paquetes se ordenan por ser “P2W” o no y luego por su orden de aparición.

La entrada debe ser leída de forma estándar.

Salida

Por cada camión que sale tienes que imprimir los identificadores de los paquetes que se va a llevar el camión (sin rebasar su capacidad), separados por un espacio. El camión puede salir sin llenarse e incluso vacío si no hay paquetes suficientes esperando en la cola. Los paquetes se identifican por su orden de aparición en la entrada, empezando en el 0.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
8 NOR NOR P2W NOR R 2 P2W R 5 NOR	2 0 4 1 3

Límites

- $1 \leq N \leq 200000$
- $1 \leq C \leq 100$

Problema D

Desfase horario

Durante la cuarentena todos los hábitos que realizábamos se tuvieron que pausar. Esto provocó un desfase horario en muchas personas, provocando no poder dormir, tener un aumento de pesadillas, no sabiendo como organizarse, etc. Como buenos programadores, no tenemos miedo a organizarnos. Tenemos todas las tareas con sus diferentes horarios, ya que no somos multihilo (todavía) no podremos realizar dos tareas que solapen el instante de tiempo. Queremos hacer un programa que nos diga que tareas debemos realizar para quitarnos la mayor cantidad de ellas.

Input

La primera línea contendrá un número N , denotando el número de tareas que tenemos que realizar. Seguidamente vendrán N líneas, cada línea mostrará el nombre de la tarea junto a dos números, el instante de tiempo de inicio I y el instante de finalización F (ambos incluidos, es decir si tenemos una tarea que empieza en 1 y acaba en 5 el tiempo total es 6, si una tarea comenzase en 5, solaparía con la anterior y se debería decidir entre una de ellas).

La entrada debe ser leída de forma estándar.

Output

La salida se corresponderá con un número denotando el máximo de tareas que podemos realizar.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
5 EstudiarAlgoritmos 20 30 Respirar 35 40 Comer 31 60 Respirar 10 15 SacarAlPerro 80 100	4

Constraints

- $1 \leq N \leq 1000$
- $1 \leq F, I \leq 10000$
- Nombre de la tarea es una secuencia del alfabeto inglés con menos de 20 caracteres sin espacios.

Problema E

Agrupación de elementos

David es un profesor de literatura que tiene unos 3275 libros escritos, bueno, igual no tantos. Pero algunos sí que tiene, su pasatiempo es escribir, le apasionan las letras, tanto, que ha decidido jugar un nuevo juego con su sobrino, que también se llama David.

David le ordena a David que escriba un conjunto de caracteres, tan largo como él quiera, y al lado de la cadena formada, un número K , David entonces intentará separar la cadena en subcadenas tal que no contengan, en cada subcadena, más de K caracteres diferentes.

Esta tarea es muy fácil hacerla con cadenas pequeñas, por ejemplo, si David escribe “palabra” y $K = 2$, una posible respuesta podría ser separarlas en las subcadenas “p”, “ala”, “br”, “a”.

Dado que el máximo número de cadenas que puede hacer sería siempre separar letra a letra, estamos interesados en saber cuál es el mínimo número de cadenas tal que una subcadena dentro de esa cadena, no supere K distintos símbolos.

Entrada

Una lista de cadenas, separadas en cada línea, denotando las palabras que se deben procesar, después de cada palabra seguirá un número K , denotando la mayor cantidad de caracteres distintos que puede tener una subcadena.

La entrada debe ser leída de forma estándar.

Salida

Para cada caso debes imprimir un número con la menor cantidad de subcadenas posibles que contengan hasta K caracteres distintos.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
palabra 2	4
aaaaaa 12	1
abba 3	1
dancingqueen 2	6

Límites

- $1 \leq S \leq 1,000,000$
- Todas las letras en S están en minúsculas en alfabeto inglés
- $1 \leq K \leq 26$