

A - Maps-STL

Tú eres el asistente de un profesor en la escuela y ella esta corrigiendo pruebas de alumnos. Cada alumno tiene multiples hojas de respuestas. Entonces la profesora quiere hacer Q consultas.

- 1 X Y: agrega el puntaje Y al estudiante con nombre X.
- 2 X: elimina el puntaje del estudiante con nombre X.
- 3 X: muestra por pantalla el puntaje del estudiante con nombre X. (Si X no tiene puntaje, entonces muestre por pantalla 0)

Input

La primera línea del input contiene un entero Q , donde este es el número de consultas. Las siguientes Q líneas contienen una consulta cada una. El primer entero, *type* de cada consulta, es el tipo de la consulta. Si la consulta es de tipo 1, esta consta de un string X y un entero Y , donde X es el nombre del alumno e Y las calificaciones del alumno. Si la consulta es de tipo 2 o 3, consta de un solo string, correspondiente al nombre del alumno.

Restricciones

- $1 \leq Q \leq 10^5$
- $1 \leq type \leq 3$
- $1 \leq X.length() \leq 6$
- $1 \leq Y \leq 10^3$

Output

Por cada consulta de tipo 3, muestre por pantalla el puntaje del estudiante dado.

Ejemplo

Input

```
1 7
2 1 Jesse 20
3 1 Jess 12
4 1 Jess 18
5 3 Jess
6 3 Jesse
7 2 Jess
8 3 Jess
```

Output

```
1 30
2 20
3 0
```

B - Radio Station

Mientras los chicos freían las instalaciones de la estación de radio, el director de la escuela les dio tareas como castigo. La tarea de Dustin era agregar comentarios a la configuración de nginx para el sitio web de la escuela. La escuela tiene n servidores. Cada servidor tiene un nombre y una ip (los nombres no son necesariamente únicos, pero los ips sí lo son). Dustin conoce la ip y el nombre de cada servidor. Para simplificar, asumiremos que un comando nginx tiene la forma "comando ip;" donde comando es una cadena que consta de letras minúsculas en inglés únicamente, e ip es la ip de uno de los servidores de la escuela.

Cada ip tiene la forma "a.b.c.d" donde a , b , c y d son números enteros no negativos menores o iguales a 255 (sin ceros a la izquierda). El archivo de configuración nginx al que Dustin debe agregar comentarios tiene comandos m . Nadie memoriza las ips de los servidores, así que para entender mejor la configuración, Dustin tiene que comentar el nombre del servidor al que pertenece la ip al final de cada línea (después de cada comando). Más formalmente, si una línea es "comando ip;" Dustin tiene que reemplazarlo con "comando ip; #nombre" donde nombre es el nombre del servidor con ip igual a ip.

Dustin no sabe nada sobre nginx, así que volvió a entrar en pánico y sus amigos te pidieron que hicieras su tarea por él.

Input

La primera línea de entrada contiene dos números enteros n y m ($1 \leq n, m \leq 1000$).

Las siguientes n líneas contienen los nombres y las ips de los servidores. Cada línea contiene un string de nombre, el nombre del servidor y un string de ip, ip del servidor, separados por espacio ($1 \leq |\text{nombre}| \leq 10$, el nombre solo consta de letras minúsculas en inglés). Se garantiza que todas las direcciones ip son distintas.

Output

Muestre por pantalla m líneas, los comandos en el archivo de configuración después de que Dustin hizo su tarea.

Ejemplo

Input

```
1 2 2
2 main 192.168.0.2
3 replica 192.168.0.1
4 block 192.168.0.1;
5 proxy 192.168.0.2;
```

Output

```
1 block 192.168.0.1; #replica
2 proxy 192.168.0.2; #main
```

Input

```
1 3 5
2 google 8.8.8.8
3 codeforces 212.193.33.27
4 server 138.197.64.57
5 redirect 138.197.64.57;
6 block 8.8.8.8;
7 cf 212.193.33.27;
8 unblock 8.8.8.8;
9 check 138.197.64.57;
```

Output

```
1 redirect 138.197.64.57; #server
2 block 8.8.8.8; #google
3 cf 212.193.33.27; #codeforces
4 unblock 8.8.8.8; #google
5 check 138.197.64.57; #server
```

C - Letter

Vasya decidió escribir una carta anónima cortando las letras del encabezado de un periódico. Conoce el encabezado s_1 y el texto s_2 que quiere enviar. Vasya puede usar cada letra de encabezado no más de una vez. Vasya no tiene que eliminar los espacios del encabezado, simplemente deja un espacio en blanco para marcarlos. Ayúdalo a averiguar si se las arreglar para componer el texto que quiere enviar.

Input

La primera línea contiene un encabezado de periódico s_1 . La segunda línea contiene el texto de la letra s_2 . s_1 y s_2 son líneas no vacías formadas por espacios, letras latinas mayúsculas y minúsculas, cuya longitud no supera los 200 caracteres. Se deben diferenciar las letras mayúsculas y minúsculas. Vasya no elimina espacios del encabezado.

Output

Si Vasya puede escribir la carta anónima, muestre por pantalla "YES", si no muestre por pantalla "NO".

Ejemplo

Input

```
1 Instead of dogging Your footsteps it disappears but you dont notice anything
2 where is your dog
```

Output

```
1 NO
```

```
1 Instead of dogging Your footsteps it disappears but you dont notice anything
2 Your dog is upsteairs
```

Output

```
1 YES
```

```
1 Instead of dogging your footsteps it disappears but you dont notice anything
2 Your dog is upsteairs
```

Output

```
1 NO
```

```
1 abcdefg hijk
2 k j i h g f e d c b a
```

Output

1 YES

D - Sum of two values

Se le da un arreglo de n enteros y su tarea es encontrar dos valores (en posiciones distintas) cuya suma sea x .

Input

La primera línea del input tiene dos números enteros n y x : el tamaño del arreglo y la suma objetivo.

La segunda línea contiene n enteros a_1, a_2, \dots, a_n : los valores del arreglo.

Restricciones

- $2 \leq n \leq 2 \cdot 10^5$
- $1 \leq x, a_i \leq 10^9$

Output

Muestre por pantalla dos enteros: la posiciones de los valores. Si hay múltiples soluciones, puede mostrar por pantalla cualquiera de esas soluciones. Si no hay soluciones, muestra por pantalla -1

Ejemplo

Input

```
1 4 8
2 2 7 5 1
```

Output

```
1 2 4
```