

Luchando contra la Hidra

Contribución de Agustín Santiago Gutiérrez

Descripción del problema

El valiente Hércules desea derrotar a la Hidra de Lerna. Tiene en su poder N espadas, numeradas desde 0 hasta $N - 1$ inclusive. Cada una de las espadas posee diferentes características.

La característica más importante de la Hidra es que posee inicialmente H cabezas, pero esto irá cambiando durante el combate. El combate se da por turnos, y en cada turno, Hércules elige exactamente una espada para utilizar en ese turno.

La espada i -ésima puede utilizarse como máximo c_i veces durante el combate, pues luego ya se rompe por el uso excesivo. Si en un turno Hércules utiliza la espada i -ésima, destruye inmediatamente d_i cabezas de la hidra. Si al hacer esto la hidra se queda sin cabezas (porque tenía d_i o menos cabezas en ese momento del combate), la lucha termina y Hércules sale airoso. En caso contrario, la hidra rápidamente regenera r_i cabezas nuevas en ese mismo turno, sin darle tiempo antes a Hércules para utilizar otra (o la misma) espada. No hay límite superior a la cantidad de cabezas que puede llegar a tener la Hidra durante el combate, e incluso puede llegar a tener mucho más que las H cabezas iniciales.

Hércules es muy poderoso e inteligente, pero no sabe programar. Tu tarea consiste en implementar una función que dados los valores H, c_i, d_i, r_i calcule la mínima cantidad de turnos en los que Hércules puede derrotar a la Hidra, y le indique un plan de espadas a utilizar para alcanzar este mínimo número de turnos.

Si hay más de una forma posible de derrotar a la hidra en la mínima cantidad de turnos, cualquiera sirve. Si no fuera posible derrotar a la Hidra, también debes indicarlo.

Descripción de la función

Debes implementar la función `hercules(H, c, d, r, e, u)`, que recibe:

- H : La cantidad H de cabezas iniciales de la Hidra.
- c, d, r : Arreglos de N enteros cada uno, con los valores c_i, d_i, r_i para cada espada.
- e, u : Arreglos de enteros en los que se debe almacenar una descripción de las espadas que debe utilizar Hércules en cada turno. Ambos deberán tener la misma longitud I , que no deberá ser más que 500.000. Estos arreglos describen que, para cada i en orden desde 0 hasta $I - 1$ inclusive, a continuación se debe utilizar la espada $e[i]$ durante los próximos $u[i]$ turnos. Si no es posible derrotar a la Hidra, se deben dejar vacíos.

La función debe retornar la mínima cantidad de turnos necesarios para derrotar a la Hidra. Si no es posible hacerlo, se debe retornar -1 .

Evaluador

El evaluador local lee de la entrada estándar:

- Una línea con dos enteros N y H
- N líneas con los enteros c_i, d_i, r_i

Escribe a la salida estándar una primera línea con el valor retornado por la función, y luego dos líneas más con los valores almacenados en los parámetros e y u , en ese orden.

Restricciones

- $1 \leq N \leq 200.000$
- $1 \leq H \leq 10^{18}$
- $1 \leq c_i, d_i, r_i \leq 10^9$

Ejemplo

Si el evaluador recibiera:

3	100
1	50 1
5	30 8
10000	1 50000

Una posible salida correcta es:

3
0 1
1 2

Que corresponde a usar primero la espada **0** durante **1** turno, y luego la espada **1** durante **2** turnos.

Otra salida también correcta es:

3
0 1 1
1 1 1

Que en este caso describe la misma solución, pero de un modo diferente.

Puntuación

Se obtiene el 25% del puntaje por el valor de retorno correcto, y el 75% restante por además dar un resultado correcto en los arreglos e, u .

Subtareas

1. $N, H, d_i, r_i \leq 1000, c_i = 1$ (4 puntos)
2. $H, c_i, d_i, r_i \leq 1000$ (8 puntos)
3. $N \leq 1000$ (12 puntos)
4. $c_i = 1$ (24 puntos)
5. Sin más restricción (52 puntos)