Horas de cierre

Hungría es un país con N ciudades, enumeradas de 0 a $N-1$.

Las ciudades están conectadas por $N-1$ carreteras *bidireccionales*, enumerados de 0 a $N-2$. Para cada j tal que $0 \leq j \leq N-2$, la carretera j conecta a la ciudad $U[j]$ y la ciudad $V[j]$ con una longitud de $W[j]$, es decir, permite que alguien viaje entre las ciudades en $W[j]$ unidades de tiempo. Cada carretera conecta dos ciudades distintas, y cada par de ciudades están conectadas por una sola carretera.

Un **recorrido** entre dos ciudades distintas a y b es una secuencia $p_0, p_1, \ldots, p_t$ de diferentes ciudades, tal que:

- $p_0 = a$,
- $p_t = b$,
- para cada i ($0 \leq i < t$), hay una carretera conectando las ciudades p_i y p_{i+1}.

Es posible viajar de cualquier ciudad a cualquier otra ciudad usando las carreteras, es decir, existe un recorrido entre cada 2 ciudades distintas. Se puede demostrar que este recorrido es único para cada par de ciudades distintas.

La **longitud** de un recorrido $p_0, p_1, \ldots, p_t$ es la suma de las longitudes de las t carreteras conectando ciudades consecutivas a lo largo del recorrido.

En Hungría, mucha gente viaja para asistir a las festividades del Día de la Fundación llevadas a cabo en dos grandes ciudades. Una vez las celebraciones terminan, las personas regresan a sus hogares. El gobierno quiere evitar que las multitudes molesten a los locales, así que planean cerrar todas las ciudades a ciertas horas. A cada ciudad se le asignara una **hora de cierre** no negativa, dada por el gobierno. El gobierno ha decidido que la suma de todas las horas de cierre no debe ser mayor que K. Más precisamente, para todo i entre 0 y $N-1$, inclusive, la hora de cierre asignada a la ciudad i es un entero no negativo $c[i]$. La suma de todos los valores $c[i]$ no debe ser mayor que K.

Considera una ciudad a y alguna asignación de horas de cierre. Se dice que una ciudad b es alcanzable desde la ciudad a si y solo si se cumple que $b = a$, o que el recorrido $p_0, \ldots, p_t$ entre estas dos ciudades (en particular $p_0 = a$ y $p_t = b$) satisface las siguientes condiciones:

- la longitud del recorrido p_0, p_1 es a lo mucho $c[p_1]$, y
- la longitud del recorrido p_0, p_1, p_2 es a lo mucho $c[p_2]$, y

- $\ldots$
- la longitud del recorrido $p_0, p_1, p_2, \ldots, p_t$ es a lo mucho $c[p_t]$.

Este año, los sitios seleccionados para el festival están localizados en las ciudades X y en la ciudad Y. Para cada asignación de horas de cierre, el **puntaje de conveniencia** es definido como la suma de los siguientes dos números:

- El número de ciudades alcanzables desde la ciudad X.
- El número de ciudades alcanzables desde la ciudad Y.

Notar que si una ciudad es alcanzable desde la ciudad X y también es alcanzable desde la ciudad Y, cuenta *doble* para el puntaje de conveniencia.

Tu tarea es encontrar el máximo puntaje de conveniencia posible que puede ser alcanzado usando alguna asignación de horas de cierre.

Detalles de Implementación

Debes implementar la siguiente función

```
int max_score(int N, int X, int Y, int64 K, int[] U, int[] V, int[] W)
```

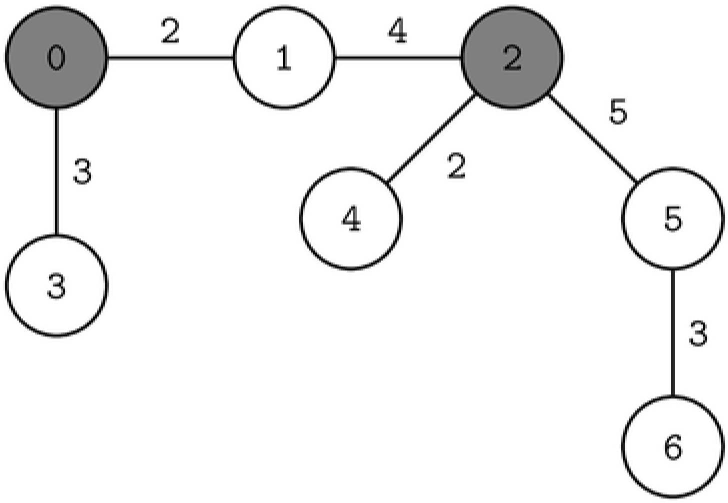
- N: el número de ciudades.
- X, Y: las ciudades con las festividades.
- K: el límite superior para la suma de horas de cierre.
- U, V: arreglos de longitud $N-1$ describiendo las conexiones de las carreteras.
- W: arreglo de longitud $N-1$ describiendo las longitudes de las carreteras.
- Esta función debe retornar el máximo puntaje de conveniencia posible que puede ser alcanzado usando alguna asignación de horas de cierre.
- Esta función puede ser llamada **varias veces** en cada caso de prueba.

Ejemplo

Considera la siguiente llamada a la función:

```
max_score(7, 0, 2, 10,
          [0, 0, 1, 2, 2, 5], [1, 3, 2, 4, 5, 6], [2, 3, 4, 2, 5, 3])
```

Corresponde a la siguiente red de carreteras:



Suponiendo que las horas de cierre son asignadas de la siguiente forma:

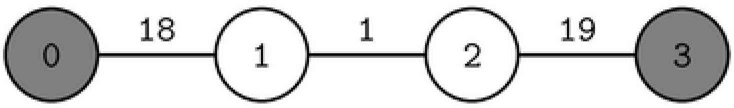
Ciudad	0	1	2	3	4	5	6
Hora de cierre	0	4	0	3	2	0	0

Nótese que la suma de todas las horas de cierre es 9, lo cual no es mayor a $K = 10$. Las ciudades 0, 1, y 3 son alcanzables desde la ciudad X ($X = 0$), mientras que las ciudades 1, 2, y 4 son alcanzables desde la ciudad Y ($Y = 2$). Por lo tanto, el puntaje de conveniencia es de $3 + 3 = 6$. No hay ninguna asignación de horas de cierre posible que resulte en un puntaje de conveniencia mayor a 6, por lo tanto, la función debería retornar 6.

También considera la siguiente llamada a la función:

```
max_score(4, 0, 3, 20, [0, 1, 2], [1, 2, 3], [18, 1, 19])
```

Corresponde a la siguiente red de carreteras:



Suponiendo que las horas de cierre son asignadas de la siguiente forma:

Ciudad	0	1	2	3
Hora de cierre	0	1	19	0

La ciudad 0 es alcanzable desde la ciudad X ($X = 0$), mientras que las ciudades 2 y 3 son alcanzables desde la ciudad Y ($Y = 3$). Por lo tanto, el puntaje de conveniencia es de $1 + 2 = 3$. No hay ninguna asignación de horas de cierre posible que resulte en un puntaje de conveniencia mayor a 3, por lo tanto, la función debería retornar 3.

Límites

- $2 \leq N \leq 200\,000$
- $0 \leq X < Y < N$
- $0 \leq K \leq 10^{18}$
- $0 \leq U[j] < V[j] < N$ (para todo j tal que $0 \leq j \leq N - 2$)
- $1 \leq W[j] \leq 10^6$ (para todo j tal que $0 \leq j \leq N - 2$)
- Es posible viajar desde cualquier ciudad a cualquier otra ciudad usando las carreteras.
- $S_N \leq 200\,000$, donde S_N es la suma de N sobre todas las llamadas a la función `max_score` en cada caso de prueba.

Subtareas

Se dice que una red de carreteras es **lineal** si la carretera i conecta las ciudades i y $i + 1$ (para cada i tal que $0 \leq i \leq N - 2$).

1. (8 puntos) La longitud del recorrido desde la ciudad X a la ciudad Y es más grande que $2K$.
2. (9 puntos) $S_N \leq 50$, la red de carreteras es lineal.
3. (12 puntos) $S_N \leq 500$, la red de carreteras es lineal.
4. (14 puntos) $S_N \leq 3\,000$, la red de carreteras es lineal.
5. (9 puntos) $S_N \leq 20$
6. (11 puntos) $S_N \leq 100$
7. (10 puntos) $S_N \leq 500$
8. (10 puntos) $S_N \leq 3\,000$
9. (17 puntos) Sin restricciones adicionales.

Evaluador de ejemplo

El valor de C denota el número de escenarios, es decir, el número de llamadas a la función `max_score`. El evaluador de ejemplo leerá la entrada siguiendo el formato:

- línea 1: C

Las descripciones de los C escenarios siguen.

El evaluador de ejemplo leerá la descripción de cada escenario siguiendo el formato:

- línea 1: $N\ X\ Y\ K$
- línea $2 + j$ ($0 \leq j \leq N - 2$): $U[j]\ V[j]\ W[j]$

El evaluador de ejemplo imprimirá una sola línea para cada escenario, siguiendo el formato:

- línea 1: el valor que retorna la función `max_score`