Tren

En el año 2992, la mayoría de los trabajos han sido ocupados por robots. Por lo tanto, mucha gente tiene abundante tiempo libre, al igual que tu familia, por lo que han decidido emprender un viaje interestelar.

Hay N planetas alcanzables numerados del 0 al $N-1$ y existen M rutas de trenes interestelares. La ruta del tren i $(0 \leq i < M)$ comienza desde el planeta $X[i]$ en el momento $A[i]$, llegando al planeta $Y[i]$ en el momento $B[i]$, con costo $C[i]$. Los trenes son el único medio de transporte entre planetas, por lo que sólo puedes bajarte de un tren en su planeta de destino, y debes tomar el siguiente tren en el mismo planeta (los transbordos entre trenes no toman tiempo). Formalmente, una secuencia de trenes $q[0], q[1], \ldots, q[P]$ es válida de tomarse si y sólo si para cualquier $1 \leq k \leq P$, $Y[q[k-1]] = X[q[k]]$ y $B[q[k-1]] \leq A[q[k]]$.

Como los viajes interestelares consumen mucho tiempo, te das cuenta de que además del boleto de tren, el costo de las comidas es importante. Afortunadamente, **los trenes interestelares ofrecen comida ilimitada y gratuita**. Pero mientras tu familia espera el próximo tren en cualquier planeta i, tienes que pagar por cada comida al costo $T[i]$.

Tu familia necesita tener W comidas, y la i-ésima $(0 \leq i < W)$ comida se puede tomar **instantáneamente** en cualquier momento entre $L[i]$ y $R[i]$ **(inclusive)**.

Ahora en el momento 0, tu familia está en el planeta 0. Necesitas calcular el costo mínimo para llegar al planeta $N-1$. Si no puedes llegar allí, tu respuesta debería ser -1.

Detalles de Implementación

Debes implementar la siguiente función:

```
long long solve(int N, int M, int W, std::vector<int> T,
                std::vector<int> X, std::vector<int> Y,
                std::vector<int> A, std::vector<int> B, std::vector<int> C,
                std::vector<int> L, std::vector<int> R);
```

Donde:

- N: El número de planetas.
- M: El número de rutas de trenes interestelares.

- W: El número de comidas.
- T: Un arreglo con una longitud de N. $T[i]$ representa el costo de cada comida en el planeta i.
- X, Y, A, B, C: cinco arreglos con una longitud de M. La tupla $(X[i], Y[i], A[i], B[i], C[i])$ describe la i-ma ruta de tren.
- L, R: Dos arreglos con un tamaño de W. El par $(L[i], R[i])$ describe el intervalo de tiempo para tener la i-ma comida.
- Esta función debería devolver el costo mínimo para llegar al planeta $N-1$ desde el planeta 0 si puedes llegar al planeta $N-1$, y -1 si no es posible.
- Para cada caso de prueba, esta función se llamará exactamente una vez.

Ejemplos

Ejemplo 1

Considera la siguiente llamada:

```
solve(3, 3, 1, {20, 30, 40}, {0, 1, 0}, {1, 2, 2},
        {1, 20, 18}, {15, 30, 40}, {10, 5, 40}, {16}, {19});
```

Una forma de llegar al planeta N es tomando el tren 0 y luego el tren 1, lo cual cuesta 45 (el cálculo detallado se muestra a continuación).

Tiempo	Acción	Costo (si corresponde)
1	Tomas el tren 0 en el planeta 0	10
15	Llegas al planeta 1	
16	Tomas la comida 0 en el planeta 1	30
20	Tomas el tren 1 en el planeta 1	5
30	Llegas al planeta 2	

Una mejor manera de llegar al planeta N es tomando el tren 2 solamente, lo cual cuesta 40 (el cálculo detallado se muestra a continuación).

Tiempo	Acción	Costo (si corresponde)
18	Tomas el tren 2 en el planeta 0	40
19	Tomas la comida 0 en el Tren 2	
40	Llegas al planeta 2	

Por lo tanto, la función debe devolver 40.

Ejemplo 2

Considera la siguiente llamada:

```
solve(3, 10, 6, {30, 38, 33}, {0, 1, 0, 0, 1}, {2, 0, 1, 2, 2},
        {12, 48, 26, 6, 49}, {16, 50, 28, 7, 54}, {38, 6, 23, 94, 50},
        {32, 14, 42, 37, 2, 4}, {36, 14, 45, 40, 5, 5});
```

El camino óptimo es tomar el tren 0 con un costo de 38. La comida 1 se puede tomar gratis en el tren 0. Las comidas 0, 2, y 3 se pueden tomar en el planeta 2 por $33 \times 3 = 99$. Las comidas 4 y 5 se pueden tomar en el en el planeta 0 por $30 \times 2 = 60$. El costo total es $38 + 99 + 60 = 197$.

Por lo tanto, la función debe regresar 197.

Límites

- $2 \leq N \leq 10^5$.
- $0 \leq M, W \leq 10^5$.
- $0 \leq X[i], Y[i] < N, X[i] \neq Y[i]$.
- $1 \leq A[i] < B[i] \leq 10^9$.
- $1 \leq T[i], C[i] \leq 10^9$.
- $1 \leq L[i] \leq R[i] \leq 10^9$.

Subtareas

1. (5 puntos): $N, M, A[i], B[i], L[i], R[i] \leq 10^3$ y $W \leq 10$.
2. (5 puntos): $W = 0$.
3. (30 puntos): No hay dos comidas que se superpongan en el tiempo. Formalmente, para cualquier momento z dónde $1 \leq z \leq 10^9$, hay como máximo una i ($0 \leq i < W$) tal que $L[i] \leq z \leq R[i]$.
4. (60 puntos): Sin restricciones adicionales.

Evaluador de Ejemplo

El evaluador de ejemplo lee la entrada en el siguiente formato:

- Línea 1: $N\ M\ W$
- Línea 2: $T[0]\ T[1]\ T[2]\ \cdots\ T[N-1]$
- Línea $3 + i$ $(0 \leq i < M)$: $X[i]\ Y[i]\ A[i]\ B[i]\ C[i]$
- Línea $3 + M + i$ $(0 \leq i < W)$: $L[i]\ R[i]$

El evaluador de ejemplo imprime tus respuestas en el siguiente formato:

- Línea 1: el valor de retorno de `solve`