

Tren

2992 yılında çoğu iş robotlar tarafından ele geçirilecek. Bu nedenle birçok insanın, ve ailenizle sizin de, çok boş vakti var ve bu nedenle yıldızlararası seyahat yapmaya karar verdiniz!

N adet ulaşılabilir gezegen vardır ve 0'dan $N - 1$ 'e numaralandırılmıştır. Ayrıca M tane yıldızlararası tren rotası vardır. i . ($0 \leq i < M$) tren rotası gezegen $X[i]$ 'den $A[i]$ anında başlar ve gezegen $Y[i]$ 'ye $B[i]$ anında ulaşır ve bu rotanın maliyeti $C[i]$ 'dir. Bir trenden sadece hedef gezegende inebilirsiniz, ayrıca yeni trene de indiğiniz gezegenden binmeniz gerekmektedir (aktarma zaman almamaktadır yani indiğiniz an binebilirsiniz). Tren rotaları dışında herhangi bir ulaşım aracı mevcut değildir. Kısaca, tren dizisi $q[0], q[1], \dots, q[P]$ ile seyahat etmek eğer bütün $1 \leq k \leq P$ olacak şekilde k 'lar için, $Y[q[k-1]] = X[q[k]]$ ve $B[q[k-1]] \leq A[q[k]]$ şartları sağlanıyorsa mümkündür.

Yıldızlararası seyahat çok zaman aldığı için tren ücretlerine ek olarak yemek ücretlerinin de çok önemli olduğunu anladınız. Neyse ki, **yıldızlararası trenler bedava yemek dağıtıyor**. Kısaca eğer i -th trendeyse, $A[i]$ ve $B[i]$ arasındaki her zamanda (**sınırlar da dahil** $[A[i], B[i]]$) istediğiniz kadar yemeği bedavaya alabiliyorsunuz. Fakat i . gezegende yeni tren beklerken, her yemek için $T[i]$ ödemek zorundasınız.

Aileniz W öğün yemeli, ve i -th ($0 \leq i < W$) öğün $L[i]$ - $R[i]$ aralığında yemek zorundadır (**sınırlar dahil**).

0. zamanda 0. gezegende bulunmaktasınız. $N - 1$. gezegene ulaşmanın minimum maliyetini yazdırmalısınız. Eğer ulaşamıyorsanız da cevap -1 olmalıdır.

Implementation Details

You need to implement the following function:

```
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);
```

- N : Gezegen sayısı.
- M : Tren rotası sayısı.
- W : Öğün sayısı.

- T : N uzunluğunda bir dizi. $T[i]$ i . gezegendeki yemek maaliyetini temsil etmektedir.
- X, Y, A, B, C : 5 adet M uzunluğunda dizi. $(X[i], Y[i], A[i], B[i], C[i])$ i -th tren rotasını tarif ediyor.
- L, R : W uzunluğunda iki dizi. $(L[i], R[i])$ i -th öğünün yenmesi gereken zaman aralığını belirtmektedir.
- Bu fonksiyon 0. gezegenden $N - 1$. gezegene gitmenin minimum maaliyetini dönmelidir. Eğer $N - 1$. düğüme ulaşmak mümkün değilse -1 dönmelidir.
- For each test case, this function will be called exactly once.

Examples

Example 1

Consider the following call:

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

One way to reach planet $N - 1$ is by taking Train 0 and then Train 1, which costs 45 (the detailed calculation is shown below).

Time	Action	Cost (if any)
1	Take Train 0 in Planet 0	10
15	Arrive in Planet 1	
16	Take meal 0 in Planet 1	30
20	Take Train 1 in Planet 1	5
30	Arrive in Planet 2	

A better way to reach planet $N - 1$ is by taking Train 2 only, which costs 40 (the detailed calculation is shown below).

Time	Action	Cost (if any)
18	Take Train 2 in Planet 0	40
19	Take meal 0 on Train 2	
40	Arrive in Planet 2	

In this way of reaching planet $N - 1$, it's also valid to take meal 0 at time 18.

Therefore, the function should return 40.

Example 2

Consider the following call:

```
solve(3, 5, 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});
```

The optimal path is to take Train 0 with a cost of 38. Meal 1 can be taken for free on Train 0. Meals 0, 2, and 3 must be taken on Planet 2 for $33 \times 3 = 99$. Meals 4 and 5 must be taken on Planet 0 for $30 \times 2 = 60$. The total cost is $38 + 99 + 60 = 197$.

Therefore, the function should return 197.

Constraints

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

Subtasks

1. (5 points): $N, M, A[i], B[i], L[i], R[i] \leq 10^3$ and $W \leq 10$.
2. (5 points): $W = 0$.
3. (30 points): No two meals overlap in time. Formally, for any time z where $1 \leq z \leq 10^9$, there is at most one i ($0 \leq i < W$) such that $L[i] \leq z \leq R[i]$.
4. (60 points): Hiçbir ek kısıt yoktur.

Sample Grader

The sample grader reads the input in the following format:

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

The sample grader prints your answers in the following format:

- Line 1: the return value of solve