

Camino de Toriis

Contribución de Lautaro Lasorsa

Descripción del problema

En su viaje por Japón, Agustín pasó por un parque lleno de toriis (arcos), como el que se muestra a continuación.



El parque estaba compuesto por N plazas y M senderos que las unían. Los senderos que unen las plazas pueden recorrerse únicamente en un sentido, especificado en el mapa del parque.

Sobre cada sendero pueden ubicarse cero o más toriis. Al transitar un sendero, una persona puede decidir **por cada torii del sendero**, si pasar por debajo de él (es decir, debajo del arco), o si pasar a su costado (no pasar por debajo del arco).

Agustín tiene una manía con los números pares, así que está interesado en recorrer el parque pasando por debajo de un **número par** de toriis en total. Específicamente, está interesado en saber de cuántas formas diferentes puede hacer esto.

Recorrer el parque significa ir desde cualquiera de las entradas hasta cualquiera de las salidas, recorriendo sólo los senderos habilitados y de la forma habilitada. Está garantizado que las únicas plazas a las que no entra ningún sendero son las entradas, que las únicas plazas de las que no sale ningún sendero son las salidas, y que no hay ninguna plaza que sea entrada y salida a la vez.

Dos formas se consideran distintas si pasan por un conjunto distinto de senderos y plazas, o si pasando por el mismo

conjunto de senderos y plazas, pasan debajo de un subconjunto distinto de toriis.

Como Agustín sabe que el número total de formas puede ser muy grande, desea saber su valor módulo 112.345.687 (número primo).

Agustín tiene el mapa del parque, que muestra las plazas, los senderos que las unen con su correspondiente sentido de recorrido, y en cada sendero un número que representa cuántos toriis hay en ese sendero.

Por seguridad, el parque está diseñado de tal forma que si una persona lo recorre, **siempre** llegará en algún momento a una salida, sin importar cuán perdida esté.

Detalles de implementación

Se debe implementar la función `toriiis(N, inicio, fin, cantidad)` que recibe la cantidad de plazas N y tres arreglos de longitud M que especifican los senderos: el sendero i -ésimo (contando desde 0), va desde la plaza `inicio[i]` hasta la plaza `fin[i]`, y contiene `cantidad[i]` toriis.

La función debe retornar un único entero, con el resto de dividir la cantidad de caminos pedida por 112.345.687.

Evaluador local

El evaluador local lee de la entrada estándar:

- Una primera línea con dos enteros N y M : la cantidad de plazas y la cantidad de senderos entre ellas.
- M líneas más: La i -ésima de estas líneas con tres enteros `inicio[i]`, `fin[i]` y `cantidad[i]`.

Devuelve por salida estándar una línea con un único entero: el valor retornado por la función `toriiis`.

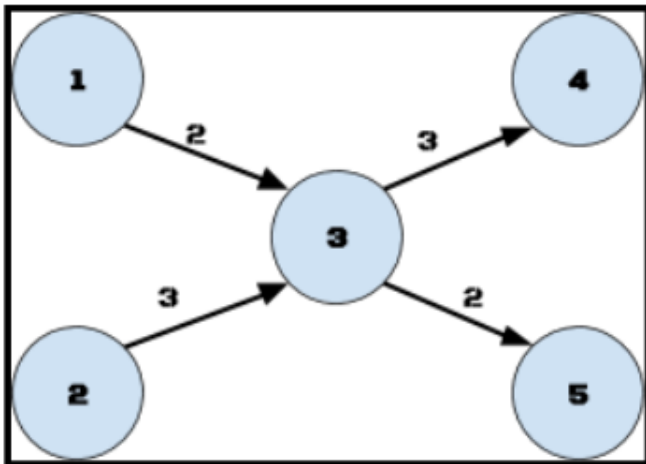
Restricciones

- $2 \leq N \leq 100.000$
- $1 \leq M \leq 200.000$
- $1 \leq inicio[i], fin[i] \leq N$
- $0 \leq cantidad[i] \leq 10^{18}$

Ejemplos

Si los datos de entrada fueran:

```
5 4
1 3 2
2 3 3
3 4 3
3 5 2
```



La respuesta correcta sería:

72

Subtareas

1. $cantidad[i] = 0$ (10 puntos)
2. $cantidad[i] \leq 5$ (15 puntos)
3. $N = 2, M = 1$ (25 puntos)
4. Sin más restricción (50 puntos)