

## C. Seguridad

time limit per test: 2 s

memory limit per test: 512 MB

Tuki está ayudando al Gobierno de la Ciudad a instalar sistemas de seguridad en las distintas calles de la Capital. Esta se modela como un multigrafo no dirigido con  $n$  nodos donde estos denotan esquinas de la ciudad mientras que los ejes representan las calles. Para cada calle se conoce su longitud y el costo de colocar un sistema de seguridad en la misma, el cual es exactamente el doble de su longitud. Como cubrir todas las calles es muy caro, el Gobierno se propone solo cubrir aquellas que pertenecen a algún camino mínimo entre el nodo 0 y el  $n - 1$ , y está interesado en descubrir el costo del proyecto.

Debemos ayudar a Tuki a resolver este problema. Para eso, debemos diseñar un algoritmo que dada la descripción de la ciudad encuentre el costo de instalar sistemas de seguridad en todas las calles que pertenecen a algún camino mínimo entre el nodo 0 y el  $n - 1$ .

### Input

La entrada consiste de una primera línea con dos enteros  $n, m$  ( $1 \leq n \leq 10000$ ,  $1 \leq m \leq 300000$ ) que denotan respectivamente la cantidad de esquinas y la cantidad de calles, seguida de  $m$  líneas con 3 enteros  $v, w$  y  $c$  indicando que hay una calle bidireccional entre los nodos  $v$  y  $w$  de largo  $c$  (con  $1 \leq c \leq 1000$ ). Puede haber más de una calle entre dos esquinas, e incluso calles que conecten una esquina consigo misma (a lo Parque Chas). Se sabe que existe por lo menos una forma de llegar de la esquina 0 a la esquina  $n - 1$ .

### Output

Se debe imprimir un único número indicando el costo de instalar un sistema de seguridad en cada calle que pertenece a algún camino mínimo entre las esquinas 0 y  $n - 1$ .

### Examples

<b>input</b>	<a href="#">Copy</a>
<pre>4 7 0 1 1 0 2 2 0 3 10 0 3 3 1 3 2 2 3 1 1 1 1</pre>	
<b>output</b>	<a href="#">Copy</a>
<pre>18</pre>	
<b>input</b>	<a href="#">Copy</a>
<pre>3 3 0 1 2 1 2 4 2 0 5</pre>	
<b>output</b>	<a href="#">Copy</a>
<pre>10</pre>	