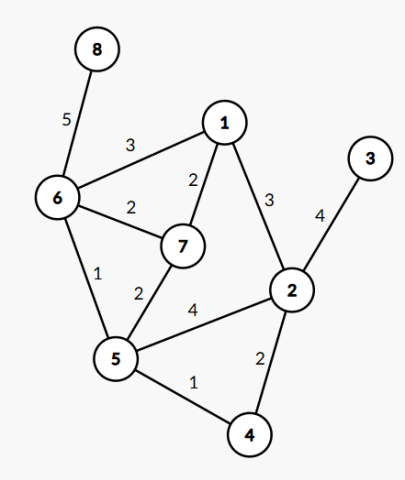
Tour de Force

August 2025
C++ — 2 SEC — 512 MB

Cycle-mania has spun Silhouettown into a frenzy, for this year the gloomy city has been chosen to host the annual cycling championship — the *Tour de Force*. Everyone, young, middle-aged, or old, has been gripped by the excitement. Bunting lines the streets, lycra is flying off the shelves, and the public water-fountains are bubbling with isotonic drinks.



There's just one snag... the organisers have yet to choose a route. It's too late to pedal back now, the peloton is coming!

To make an enjoyable yet challenging race, the *Tour de Force* route must be a loop through the city, finishing where it started, and must be as long (and gruelling) as possible. The **e** streets of Silhouettown (numbered from 1 to **e**) are connected together by **n** junctions (numbered from 1 to **n**) of varying safety. To prevent the feverish crowd from growing bored, it has been decided that no road or junction should appear more than once in the route (except the starting and ending junctions, of course).

INPUT You will be given two integers, **n** and **e**, denoting the number of junctions and streets in Silhouettown, respectively. This will be followed by **e** lines containing three integers. The **i**th such line giving **a**, **b**, and **d**, denoting that street **i** connects junctions **a** and **b** (in both directions) and has a length of **d**.

No junction will be directly connected to itself and no two junctions will be connected by more than one street. You will always be given an input containing a loop.

$3 \leq \mathbf{n}, \mathbf{e} \leq 1{,}000{,}000$

OUTPUT Output a single integer, giving the length of the longest loop in Silhouettown that does not repeat any junctions or streets (except the starting and ending junctions).

SAMPLE Suppose the streets of Silhouettown are arranged as shown above. The longest loop is highlighted in red, going $1 \to 2 \to 5 \to 7 \to 6 \to 1$, with a total length of 14.

INPUT

```
4 3
1 2 3
1 6 3
1 7 2
2 3 4
2 4 2
2 5 4
4 5 1
5 6 1
5 7 2
6 7 2
6 8 5

3 3
1 2 1
3 2 2
3 1 3
```

OUTPUT

```
14


6
```