

# F. Pająk Chwat

---

**Dostępna pamięć:** 32 MB

Pająk Chwat ma wiele ulubionych miejsc. Niektóre pary miejsc połączył nitkami, tak żeby z każdego miejsca można było przejść do każdego innego chodząc tylko po nitkach. Ponieważ do tej roboty pająk średnio się przykładął, niektóre nitki wyszły grubsze a inne cieńsze. Pomiędzy parą miejsc może istnieć więcej niż jedna nić. Krytyczną nicią danej sieci nazywamy najcieńszą jej nić.

Ostatnio pająk upasł się na tłustych muchach i postanowił przeorganizować swoją sieć, żeby lepiej utrzymywała jego wagę. Ponieważ jest leniwy, reorganizacja polega na tym, że pająk chce pousuwać (pozrywać) niektóre nici, tak żeby nadal każde miejsce było połączone (być może pośrednio) z każdym innym, ale żeby krytyczna nić powstałej w ten sposób sieci była jak najgrubsza.

## Specyfikacja danych wejściowych

W pierwszym wierszu danych wejściowych znajdują się dwie dodatnie liczby całkowite  $n$  i  $m$  oddzielone pojedynczą spacją, oznaczające odpowiednio liczbę miejsc i liczbę nici w oryginalnej sieci, gdzie  $2 \leq n \leq 10^6$  i  $1 \leq m \leq 10^6$ . Miejsca są ponumerowane od 1 do  $n$ . W każdym z kolejnych  $m$  wierszy wejścia znajdują się trzy liczby całkowite dodatnie  $a$ ,  $b$  i  $w$  oddzielone pojedynczymi spacjami, oznaczające, że w oryginalnej sieci miejsca  $a$  i  $b$  są połączone nicią o grubości  $w$  nanometrów, gdzie  $a \neq b$  oraz  $1 \leq w \leq 10^9$ .

## Specyfikacja danych wyjściowych

W pierwszym i jedynym wierszu wyjścia Twój program powinien jedną liczbę całkowitą dodatnią będącą grubością krytycznej nici w sieci zmodyfikowanej przez pająka.

### Przykład A

**Wejście:**

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

**Wyjście:**

```
3
```

### Przykład B

**Wejście:**

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

**Wyjście:**

```
2
```

### Przykład C

**Wejście:**

```
8 10
1 2 4
1 3 3
1 4 2
3 4 8
4 5 10
5 6 10
6 8 14
8 5 11
7 8 12
7 5 15
```

**Wyjście:**

```
3
```