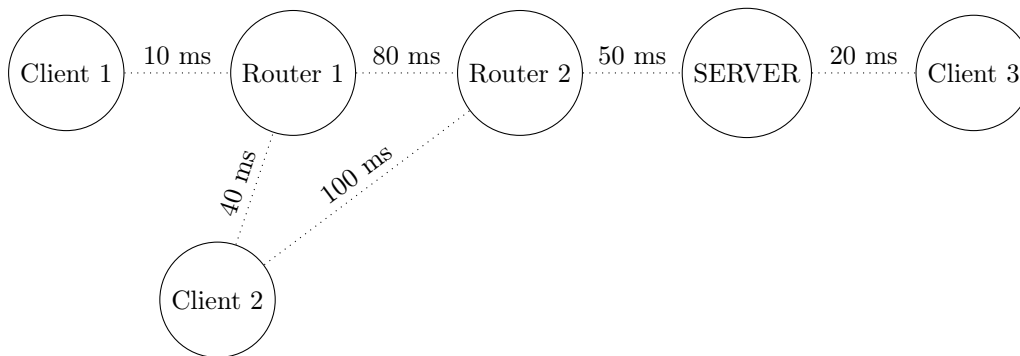


Код задачі: GMSRV

Важливим фактором для багатокористувацької онлайн-гри є низька мережева затримка від користувача до сервера. При цьому, пристрої в Інтернеті спілкуються один з одним, використовуючи мережеві маршрути, які проходять через низку проміжних вузлів-маршрутизаторів. Кожна ланка цього маршруту має власну ненульову затримку.

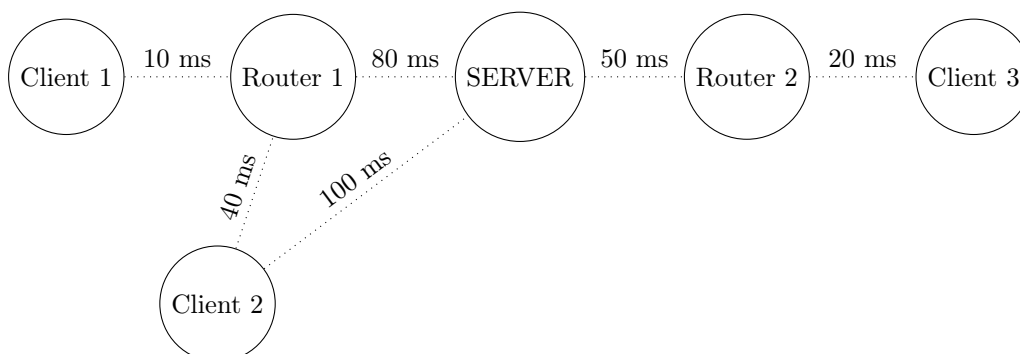


- Кожен вузол мережі може виконувати одну з трьох ролей: Client, Router або Server.
- Server може бути лише один на всю мережу.
- Усі комунікації двосторонні: якщо вузол А може спілкуватися з вузлом В, вузол В може спілкуватися з вузлом А з такою ж затримкою.
- Якщо існує кілька шляхів від клієнта до сервера, клієнт гарантовано піде шляхом з найменшою сумарною затримкою (навіть якщо цей шлях пролягає через іншого клієнта).
- Усі затримки — сталі додатні числа.

Для прикладу вище, затримки до клієнтів становлять:

- Client 1: $10 + 80 + 50 = 140$ ms
- Client 2: $100 + 50 = 150$ ms
- Client 3: 20 ms

Максимальною затримкою в такому випадку є 150 ms. Однак, якщо ми поміняємо ролями вузли “Router 2” і “Server”, затримки скоротяться до 90 ms, 100 ms і 70 ms відповідно, тоді максимальна затримка буде становити 100 ms.



Ви розробляєте онлайн-гру для користувачів зі всієї країни, і бажаєте розмістити центральний ігровий сервер таким чином, щоб максимальна затримка між сервером і кожним клієнтом була мінімальною. В якості сервера можна вибрати будь-який вузол мережі, який не є клієнтом.

Маючи інформацію про топологію мережі (які вузли з'єднані з якими, і яка затримка кожного з'єднання), знайдіть таке розташування сервера, яке мінімізує найбільше значення затримки до клієнта. Виведіть це значення затримки.

Вхідні дані

Вхідний файл `gamsrv.in` складається з $M + 2$ рядків.

- Перший рядок містить N і M — кількість вузлів та з'єднань відповідно.
 $3 \leq N \leq 1000, 2 \leq M \leq 1000$
- Другий рядок містить перелік цілих чисел, розділених пробілом — номери вузлів, які є клієнтами. Усі вузли в мережі нумеруються від 1 до N .
- Наступні M рядків містять трійки натуральних чисел *startnode*, *endnode*, *latency* — номер початкового вузла, кінцевого вузла та затримка для кожного з'єднання.
 $1 \leq latency \leq 10^9$.

Вихідні дані

Вихідний файл `gamsrv.out` повинен містити одне число — мінімальне значення найбільшої затримки до клієнта (яке ми отримаємо при оптимальному розташуванні сервера).

↓ Див. приклади нижче ↓

Приклад 1

gamsrv.in

```
6 6
1 2 6
1 3 10
3 4 80
4 5 50
5 6 20
2 3 40
2 4 100
```

gamsrv.out

```
100
```

Приклад 2

gamsrv.in

```
9 12
2 4 6
1 2 20
2 3 20
3 6 20
6 9 20
9 8 20
8 7 20
7 4 20
4 1 20
5 2 10
5 4 10
5 6 10
5 8 10
```

gamsrv.out

```
10
```

Приклад 3

gamsrv.in

```
3 2
1 3
1 2 50
2 3 1000000000
```

gamsrv.out

```
1000000000
```