

## Archi and Tree

Archi là một cậu bé ở Uzhlyandia. Trong ngày sinh nhật của mình, Chef chuẩn bị một cây gồm  $N$  đỉnh (được đánh số từ 1 tới  $N$ ). Mỗi cạnh trong cây có độ dài nhất định.

Có  $M$  quái xế trên cây, đánh số từ 1 tới  $M$ . Với mỗi  $i$ , quái xế thứ  $i$  bắt đầu vào thời điểm  $t_i$  từ đỉnh thứ  $u_i$  và chạy thẳng tới đỉnh thứ  $v_i$  với vận tốc không đổi là  $s_i$ . Trước thời điểm  $t_i$ , quái xế coi như không xuất hiện trên cây.

Chef thích thú với câu hỏi sau: với mỗi đỉnh, thời gian nhỏ nhất mà có một quái xế xuất hiện ở đỉnh đó là bao nhiêu?

### Dữ liệu vào

- Dòng đầu tiên chứa một số nguyên  $T$  thể hiện số lượng test.  $T$  test được miêu tả như sau.
- Dòng đầu tiên của mỗi test chứa một số nguyên  $N$ .
- $N-1$  dòng tiếp theo, mỗi dòng chứa ba số nguyên  $u$ ,  $v$  và  $w$  thể hiện một cạnh nối đỉnh  $u$  và  $v$  có độ dài là  $w$ .
- Dòng tiếp theo chứa một số nguyên  $M$ .
- Dòng thứ  $i$  trong  $M$  dòng tiếp theo chứa bốn số nguyên  $u_i$ ,  $v_i$ ,  $t_i$  và  $s_i$  thể hiện quái xế thứ  $i$ .

### Dữ liệu ra

- Với mỗi test, in ra  $N$  dòng. Với mỗi  $i$ , dòng thứ  $i$  chứa một số thực – thời gian nhỏ nhất mà đỉnh thứ  $i$  có quái xế đến thăm, hoặc  $-1$  nếu nó không bao giờ được đến thăm. Đáp án của bạn được cho là đúng nếu chênh lệch không quá  $10^{-6}$ .

### Ràng buộc

- $1 \leq T \leq 100$
- $1 \leq N, M \leq 2 \cdot 10^5$
- $1 \leq u, v \leq N$
- $1 \leq w, t, s \leq 10^9$

- Đồ thị được cho là một cây
- Tổng của  $N$  trong tất cả các test không vượt quá  $2 \cdot 10^5$
- Tổng của  $M$  trong tất cả các test không vượt quá  $2 \cdot 10^5$

### Subtasks

#### Subtask #1 (10 điểm):

- Tổng của  $N$  trong tất cả các test không quá  $5 \cdot 10^3$
- Tổng của  $M$  trong tất cả các test không quá  $5 \cdot 10^3$

**Subtask #2 (20 điểm):** tốc độ của các quái xé đều giống nhau

**Subtask #3 (20 điểm):** cây đã cho là một cây tre (một đường thẳng)

**Subtask #4 (50 điểm):** ràng buộc gốc

### Ví dụ

#### Input:

```
1
5
1 2 3
1 3 5
3 4 1
4 5 4
3
2 1 3 4
4 2 1 3
1 3 2 6
```

#### Output:

```
2.0000000
3.0000000
1.3333333
1.0000000
-1
```