

Динамічний діаметр (diameter)

День	1
Мова	Українська
Обмеження по часу:	5 секунд
Обмеження по пам'яті:	1024 мегабайти

Дано зважене неорієнтоване дерево з n вершинами та q оновленнями. Кожне оновлення змінює вагу одного ребра. Знайдіть та виведіть діаметр дерева після кожного оновлення.

(Відстань між двома вершинами — це сума ваг всіх ребер на унікальному простому шляху, який їх сполучає. Діаметр — це найбільша відстань між кожною парою вершин.)

Зверніть увагу, що вам потрібно рішити задачу **онлайн**.

Вхідні дані

Перший рядок містить три цілі числа n , q та w ($2 \leq n \leq 100,000, 1 \leq q \leq 100,000, 1 \leq w \leq 20,000,000,000,000$) — кількість вершин у дереві, кількість оновлень та обмеження на вагу ребер відповідно. Вершини пронумеровані від 1 до n .

Наступні $n - 1$ рядків описують початковий граф. i -й рядок містить три цілі числа a_i , b_i , c_i ($1 \leq a_i, b_i \leq n, 0 \leq c_i < w$), які означають, що спочатку було ребро між вершинами a_i та b_i з вагою c_i . Гарантується, що ці $n - 1$ рядки описують дерево.

Наступні q рядків описують запити. j -й рядок містить два цілі числа d_j , e_j ($0 \leq d_j < n - 1, 0 \leq e_j < w$). Ці два числа використовуються в цій формулі:

- $d'_j = (d_j + last) \bmod (n - 1)$
- $e'_j = (e_j + last) \bmod w$

де $last$ — це відповідь на останній запит (спочатку $last = 0$). Зверніть увагу, що вам **непотрібно** знаходити діаметр початкового графа та використовувати його значення для отримання першого запиту. Пара (d'_j, e'_j) означає, що $d'_j + 1$ -ше ребро у вхідних даних змінює вагу на e'_j .

Вихідні дані

Виведіть q рядків. Для кожного i , i -й рядок має містити діаметр дерева після i -го оновлення.

Оцінювання

Підзадача 1 (11 балів): $n, q \leq 100$ та $w \leq 10,000$

Підзадача 2 (13 балів): $n, q \leq 5,000$ та $w \leq 10,000$

Підзадача 3 (7 балів): $w \leq 10,000$ та всі ребра мають вигляд $\{1, i\}$ (Тому дерево — це зірка з центром у вершині 1.)

Підзадача 4 (18 балів): $w \leq 10,000$ та всі ребра мають вигляд $\{i, 2i\}$ або $\{i, 2i + 1\}$ (Тому, якби вершина 1 була центром, дерево було би збалансованим бінарним деревом.)

Підзадача 5 (24 бали): гарантується, що після кожного оновлення найдовший шлях проходить через вершину 1

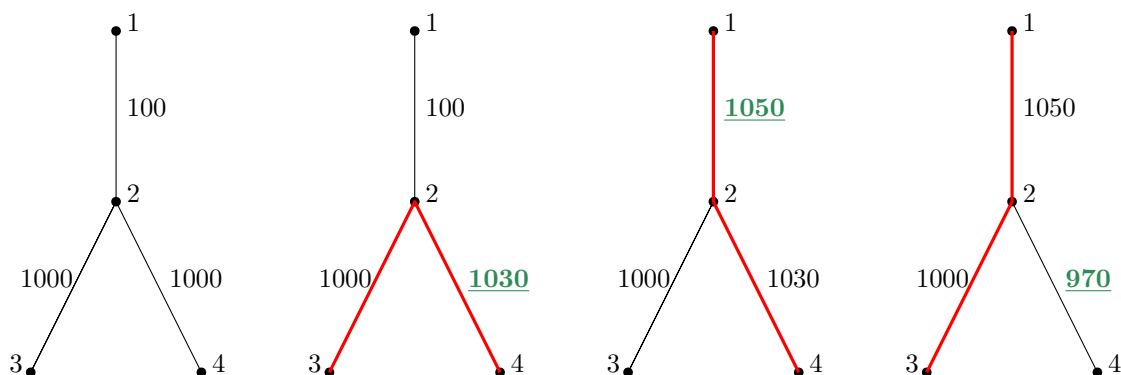
Підзадача 6 (27 балів): без додаткових обмежень

Приклади

standard input	standard output
4 3 2000 1 2 100 2 3 1000 2 4 1000 2 1030 1 1020 1 890	2030 2080 2050
10 10 10000 1 9 1241 5 6 1630 10 5 1630 2 6 853 10 1 511 5 3 760 8 3 1076 4 10 1483 7 10 40 8 2051 5 6294 5 4168 7 1861 0 5244 6 5156 3 3001 8 5267 5 3102 8 3623	6164 7812 8385 6737 6738 7205 6641 7062 6581 5155

Примітка

До першого прикладу наведені малюнки нижче. Лівий малюнок показує початковий стан графа. Кожна наступна картинка зображує граф після чергового оновлення. Вага оновленого ребра пофарбована в зелений колір, а діаметр у червоний.



Перше оновлення змінює вагу 3-го ребра, тобто $\{2, 4\}$, на 1030. Діаметр рівний 2030 — відстань між вершинами 3 та 4.

Через те, що відповідь 2030, то другий запит такий:

$$d'_2 = (1 + 2030) \bmod 3 = 0$$

$$e'_2 = (1020 + 2030) \bmod 2000 = 1050$$



Тому вага ребра $\{1, 2\}$ змінюється на 1050. Тому пара вершин $\{1, 4\}$ має найдовшу відстань між ними, а саме 2080.

Третій запит розшифровується так:

$$d'_3 = (1 + 2080) \bmod 3 = 2$$
$$e'_3 = (890 + 2080) \bmod 2000 = 970$$

Оскільки вага ребра $\{2, 4\}$ зменшилась до 970, пара вершин $\{1, 3\}$ стала мати найдовшу відстань між ними, а саме 2050.