

## Задача А. Гигантский Пингвин

|                         |                   |
|-------------------------|-------------------|
| Имя входного файла:     | стандартный ввод  |
| Имя выходного файла:    | стандартный вывод |
| Ограничение по времени: | 3 секунды         |
| Ограничение по памяти:  | 512 мегабайт      |



Pengsoo — крайне популярный гигантский Корейский пингвин. Он очень невоспитанный и любит пить.

Сейчас Pengsoo очень занят — он отвечает на запросы в очередной задаче на графы.

У него есть связный неориентированный граф, в котором каждая **вершина** лежит не более, чем на  $k$  вершинно-простых циклах.

Он хочет отвечать на запросы двух типов.

- Пометить вершину  $v$ .
- Найти ближайшую помеченную вершину для вершины  $v$  (гарантируется, что на момент данного запроса в графе имеется хотя бы одна помеченная вершина).

Pengsoo очень ленивый, поэтому он решил вздремнуть и попросил вас ответить на данные запросы. Если вы не справитесь до того, как он проснется, он будет над вами издеваться, так что поторопитесь!

### Формат входных данных

В первой строке записаны три числа  $n, m, k$  ( $1 \leq n \leq 100\,000, n-1 \leq m \leq 200\,000, 0 \leq k \leq 10$ ) — количество вершин, ребер и максимальное количество вершинно-простых циклов, проходящих через одну вершину.

В следующих  $m$  строках содержится описание ребер. В  $i$ -й строке записаны два числа  $u_i, v_i$  ( $1 \leq u_i, v_i \leq n, u_i \neq v_i$ ), означающие, что в графе есть ребро между вершинами  $u_i$  и  $v_i$ .

Гарантируется, что в графе нет красных ребер, граф является связным и каждая вершина лежит не более, чем на  $k$  вершинно-простых циклах.

В следующей строке записано число  $q$  ( $1 \leq q \leq 200\,000$ ) — количество запросов.

В каждой из следующих  $q$  строк содержится описание запроса. В  $i$ -й строке записаны два числа  $t_i, v_i$  ( $1 \leq t_i \leq 2, 1 \leq v_i \leq n$ ).

Если  $t_i = 1$ , пометьте вершину  $v_i$ . Гарантируется, что данная вершина не была помечена ранее.

Если  $t_i = 2$ , найдите расстояние до ближайшей помеченной вершины от вершины  $v_i$ . Гарантируется, что в графе уже есть хотя бы одна помеченная вершина.

### Формат выходных данных

Для каждого запроса с  $t_i = 2$  выведите расстояние до ближайшей помеченной вершины.

**Примеры**

| стандартный ввод                                                                        | стандартный вывод     |
|-----------------------------------------------------------------------------------------|-----------------------|
| 5 4 0<br>1 2<br>2 3<br>3 4<br>4 5<br>7<br>1 1<br>1 5<br>2 1<br>2 2<br>2 3<br>2 4<br>2 5 | 0<br>1<br>2<br>1<br>0 |
| 5 6 2<br>1 2<br>2 3<br>1 3<br>3 4<br>4 5<br>3 5<br>3<br>1 1<br>2 4<br>2 5               | 2<br>2                |

## Задача В. Автобус и Деревни

|                         |                          |
|-------------------------|--------------------------|
| Имя входного файла:     | <i>стандартный ввод</i>  |
| Имя выходного файла:    | <i>стандартный вывод</i> |
| Ограничение по времени: | 4 секунды                |
| Ограничение по памяти:  | 1024 мегабайта           |

В стране Байтландии  $n$  городов, соединённых  $n - 1$  двусторонней дорогой, формально Байтландия представлена деревом. Города пронумерованы целыми числами от 1 до  $n$ , дороги числами от 1 до  $n - 1$ .

Дорога номер  $i$  соединяет города  $(i + 1)$  и  $p_i$  и имеет длину  $l_i$  метров. Также вдоль  $i$ -й дороги расположено  $k_i$  деревень, расположенных на расстоянии  $0 < x_{i,1} < \dots < x_{i,k_i} < l_i$  метров от города  $i + 1$ .

В стране произвели автобус с запасом хода  $w$  метров и хотят соединить как можно больше деревень. Формально, надо выбрать две конечные точки, которые могут быть как в городе, так и на дороге. Автобус будет двигаться по единственному простому пути между этими точками. Длина этого пути должна **не превосходить**  $w$  метров. Вам требуется найти **максимальное количество деревень**, которые встретятся на пути автобуса, если его выбрать оптимально.

### Формат входных данных

В первой строке заданы два числа  $n, w$  ( $2 \leq n \leq 250000$ ) ( $1 \leq w \leq 10^{18}$ ) — кол-во городов и запас хода автобуса.

В  $i$ -й из следующих  $n - 1$  строках расположено по  $3 + k_i$  чисел:  $p_i, l_i, k_i, x_{i,1}, \dots, x_{i,k_i}$  ( $1 \leq p_i \leq i$ ) ( $0 \leq k_i \leq 10^6$ ) ( $0 < x_{i,1} < \dots < x_{i,k_i} < l_i \leq 10^{12}$ ) — описание  $i$ -й дороги.

Гарантируется, что сумма  $k_i$  не превосходит  $10^6$ .

### Формат выходных данных

Выведите единственное число — ответ на задачу.

### Примеры

| <i>стандартный ввод</i>                                                          | <i>стандартный вывод</i> |
|----------------------------------------------------------------------------------|--------------------------|
| 4 2<br>1 2 1 1<br>1 610 2 1 100<br>3 2001 0                                      | 2                        |
| 2 2<br>1 2 1 1                                                                   | 1                        |
| 8 6<br>1 2 1 1<br>1 3 2 1 2<br>2 1 0<br>3 4 1 2<br>2 3 1 1<br>1 4 1 3<br>3 4 1 1 | 4                        |

## Задача С. Очередная задача на запросы

|                         |                   |
|-------------------------|-------------------|
| Имя входного файла:     | стандартный ввод  |
| Имя выходного файла:    | стандартный вывод |
| Ограничение по времени: | 5 секунд          |
| Ограничение по памяти:  | 1024 мегабайта    |

Дано **взвешенное** дерево на  $n$  вершинах, пронумерованных целыми числами от 1 до  $n$ .

Также дана перестановка вершин  $p_1, \dots, p_n$ .

Требуется отвечать на запросы двух видов:

- 1  $l$   $r$   $v$ : Посчитать  $\sum_{i=l}^r \text{dist}(p_i, v)$ .
- 2  $i$ : поменять местами  $p_i$  и  $p_{i+1}$ .

Запросы будут **закодированы**, чтобы решение работало **онлайн**. То есть, на вход будут подаваться параметры  $\hat{i}, \hat{l}, \hat{r}$ . В то время, как реальные значения будут задаваться следующим соотношением, где  $lastans$  — ответ на последний запрос вида 1, или 0 если таких запросов ещё не было.

$$x = (lastans \bmod 2^{30}) \oplus \hat{x}$$

### Формат входных данных

В первой строке задано два числа  $n, q$  ( $2 \leq n, q \leq 2 \cdot 10^5$ ) — кол-во вершин и кол-во запросов.

Во второй строке задано  $n$  различных чисел  $p_1, \dots, p_n$ : ( $1 \leq p_i \leq n$ ) — перестановка вершин.

В следующих  $n - 1$  строках задано по три числа  $u, v, w$  ( $1 \leq u, v \leq n$ ) ( $1 \leq w \leq 10^6$ ) — описание дорог.

В следующих  $q$  строках заданы закодированные запросы ( $0 \leq \hat{l}, \hat{r}, \hat{v}, \hat{i} < 2^{30}$ ) ( $1 \leq l, r, v \leq n$ ) ( $1 \leq i < n$ ), в одном из следующих форматов:

- 1  $\hat{l}$   $\hat{r}$   $\hat{v}$
- 2  $\hat{i}$

### Формат выходных данных

Для каждого запроса типа 1 выведите требуемую сумму.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5 5              | 23                |
| 4 5 1 3 2        | 37                |
| 4 2 4            | 28                |
| 1 3 9            |                   |
| 4 1 4            |                   |
| 4 5 2            |                   |
| 1 1 5 4          |                   |
| 1 22 20 20       |                   |
| 2 38             |                   |
| 2 39             |                   |
| 1 36 38 38       |                   |