Супер Дерево

Вам дано корневое дерево с n вершинами, обозначенными индексами $0, \ldots, n-1$. Корень имеет индекс 0. Для каждого $i \in \{0, \ldots, n-1\}$ вершина i (т. е. вершина с индексом i) имеет целое число a_i, присвоенное ей. Пусть f_v — значение побитового И (далее обозначаемого $\&$) значений a_i на простом пути от вершины v к корню. (Обратите внимание, что простой путь от вершины x к вершине y включает в себя как x, так и y.) Пусть *сила* дерева равна значению

$$\sum_{0 \le u,v < n} f_u \cdot f_v,$$

и пусть *суперсила* дерева равна значению (обратите внимание на разницу в диапазонах)

$$\sum_{0 \le u < v < n} f_u \cdot f_v.$$

Поясняющий пример см. в объяснении примеров тестовых примеров ниже.

Будем говорить, что вершина u принадлежит *поддереву вершины* v, если v принадлежит простому пути от вершины u к корню. Обратите внимание, что поддерево вершины x включает в себя саму вершину x.

Вам дано q запросов обновления. Каждое обновление описывается двумя целыми числами, v и x, и требует обновления $a_u := a_u \,\&\, x$ для каждой вершины u в поддереве вершины v. После каждого обновления вы должны выводить силу и суперсилу текущего дерева.

Поскольку выходные значения могут быть большими, выведите их по модулю $10^9 + 7$.

Формат входных данных

Первая строка входных данных содержит целые числа n и q.

Вторая строка входных данных содержит $n-1$ целых чисел, а именно p_1, p_2, …, p_{n-1}, которые определяют структуру дерева. Для каждого $i \in \{1, \ldots, n-1\}$, p_i — это индекс родителя вершины i, и верно $0 \le p_i < i$.

Третья строка входных данных содержит n целых чисел, а именно a_0, a_1, …, a_{n-1}. Это значения, присвоенные вершинам.

Каждая из следующих строк q содержит два целых числа: v $(0 \leq v < n)$ и x. Эти целые числа определяют обновления.

Формат выходных данных

Выведите $q+1$ строк. Каждая строка должна содержать два целых числа, разделенных пробелом. В первой строке выведите силу и суперсилу (по модулю 10^9+7) исходного дерева. В i-й строке оставшихся q строк ($i \in \{1, \ldots, q\}$) выведите силу и суперсилу (по модулю 10^9+7) дерева после i-го обновления.

Ограничения

- $1 \leq n, q \leq 10^6$.
- $0 \leq a_i < 2^{60}$ для каждого $i \in \{0, \ldots, n-1\}$.
- $0 \leq x < 2^{60}$ для каждого обновления (v, x).

Оценивание

Для данного тестового примера ваше решение получит 50% оценки, если оно правильно вычислит все значения степени, но неправильно вычислит хотя бы одно значение сверхстепени для этого тестового примера.

Аналогично, 50% от суммы баллов за данный тестовый пример будет присуждено решению, которое правильно вычисляет все значения сверхстепени для этого тестового примера, но неправильно вычисляет хотя бы одно значение степени.

Подзадачи

1. (4 балла) $n = 3$.
2. (7 баллов) $n, q \leq 700$.
3. (13 баллов) $n, q \leq 5000$.
4. (6 баллов) $n \leq 10^5$, $p_i = i-1$ (для каждого $i \in \{1, \ldots, n-1\}$) и $a_i, x < 2^{20}$ (для каждого $i \in \{0, \ldots, n-1\}$ и для каждого обновления (v, x)).
5. (7 баллов) $p_i = i-1$ (для каждого $i \in \{1, \ldots, n-1\}$).
6. (12 баллов) $a_i, x < 2^{20}$ (для каждого $i \in \{0, \ldots, n-1\}$ и для каждого обновления (v, x)).
7. (14 баллов) $n \leq 10^5$.
8. (11 баллов) $n \leq 5 \cdot 10^5$.
9. (26 баллов) Никаких дополнительных ограничений.

Пример 1

Стандартный ввод

```
3 3
0 0
7 3 4
1 6
2 2
0 3
```

Стандартный вывод

```
196 61
169 50
81 14
25 6
```

Пояснение к примеру

Изначально у нас есть

$$f_0 = 7,\ f_1 = 7\&3 = 3,\ f_2 = 7\&4 = 4.$$

Следовательно, сила дерева равна

$$f_0 \cdot f_0 + f_0 \cdot f_1 + f_0 \cdot f_2 + f_1 \cdot f_0 + f_1 \cdot f_1 + f_1 \cdot f_2 + f_2 \cdot f_0 + f_2 \cdot f_1 + f_2 \cdot f_2 =$$

$$= 7 \cdot 7 + 7 \cdot 3 + 7 \cdot 4 + 3 \cdot 7 + 3 \cdot 3 + 3 \cdot 4 + 4 \cdot 7 + 4 \cdot 3 + 4 \cdot 4 = 196.$$

Суперсила равна

$$f_0 \cdot f_1 + f_0 \cdot f_2 + f_1 \cdot f_2 = 7 \cdot 3 + 7 \cdot 4 + 3 \cdot 4 = 61.$$

После первого обновления:

$$a_0 = 7,\ a_1 = 3\&6 = 2,\ a_2 = 4;$$

$$f_0 = 7,\ f_1 = 2,\ f_2 = 4.$$

После второго обновления:

$$a_0 = 7,\ a_1 = 2,\ a_2 = 4\&2 = 0;$$

$$f_0 = 7,\ f_1 = 2,\ f_2 = 0.$$

После третьего обновления:

$$a_0 = 7\&3 = 3,\ a_1 = 2\&3 = 2,\ a_2 = 0\&3 = 0;$$

$$f_0 = 3,\ f_1 = 2,\ f_2 = 0.$$

Пример 2

Стандартный ввод

```
4 2
0 0 1
6 5 6 2
1 2
0 3
```

Стандартный вывод

```
256 84
144 36
16 4
```

Объяснение

Изначально, мы имеем

$$f_0 = 6,\ f_1 = 6\&5 = 4,\ f_2 = 6\&6 = 6,\ f_3 = 2\&5\&6 = 0.$$

После первого обновления:

$$a_0 = 6,\ a_1 = 5\&2 = 0,\ a_2 = 6,\ a_3 = 2\&2 = 2;$$

$$f_0 = 6,\ f_1 = 0,\ f_2 = 6,\ f_3 = 2\&0 = 0.$$

После второго обновления:

$$a_0 = 7,\ a_1 = 2,\ a_2 = 4\&2 = 0;$$

$$f_0 = 7,\ f_1 = 2,\ f_2 = 0.$$

Пример 3

Стандартный ввод

```
7 3
0 0 1 1 2 2
7 6 5 7 3 4 2
4 4
3 3
2 1
```

Стандартный вывод

```
900 367
784 311
576 223
256 83
```