Супер-дрво

Дадено ви е кореново дрво (дрво со корен, анг. rooted tree) со n темиња, нумерирани со редните броеви $0, \ldots, n-1$. Коренот има реден број 0. За секое $i \in \{0, \ldots, n-1\}$, на темето i (т. е. на темето со реден број i) му е придружен цел број a_i. Нека f_v е вредноста на резултатот од операцијата логичко И на ниво на битови (во понатамошниот дел од текстот ќе биде означувана со $\&$) над вредностите a_i на простиот пат од темето v до коренот (да забележиме дека простиот пат од дадено теме x до дадено теме y ги вклучува и темињата x и y). *Моќта* на дрвото ќе ја дефинираме како вредноста на

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

а *супер-моќта* на дрвото како вредноста на:

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

(обрнете внимание на разликата во опсезите во двете формули)

Како пример за појаснување, погледнете го објаснувањето на примерите за тест случаи дадени подоле.

Ќе велиме дека темето u припаѓа на *поддрвото на темето* v ако v припаѓа на простиот пат од темето u до коренот. Да забележиме дека поддрвото на темето x го вклучува и самото теме x.

Дадени ви се q ажурирања (анг. updates). Секое ажурирање е опишано со два цели броја, v и x, и од вас побарува да поставите $a_u := a_u \,\&\, x$ за секое теме u во поддрвото на темето v. После секое ажурирање, треба да ја отпечатите моќта и супер-моќта на тековното (моменталното) дрво.

Бидејќи излезните вредности може да бидат големи, отпечатете ги по модул $10^9 + 7$.

Формат на влезот

Првата линија од влезот ги содржи целите броеви n и q.

Втората линија од влезот содржи $n-1$ цели броеви: $p_1, p_2, \ldots, p_{n-1}$, кои ја определуваат структурата на дрвото. За секое $i \in \{1, \ldots, n-1\}$, p_i е редниот број на родителот на темето

i, и притоа важи дека $0 \leq p_i < i$.

Третата линија од влезот содржи n цели броеви: a_0, a_1, $\ldots$, a_{n-1}. Ова се вредностите придружени на темињата.

Секоја од следните q линии содржи по два цели броја, v ($0 \leq v < n$) и x. Овие броеви ги специфицираат индивидуалните ажурирања.

Формат на излезот

Отпечатете $q+1$ линии. Секоја линија треба да содржи по два цели броја, разделени со по едно празно место. Во првата линија, отпечатете ја моќта и супер-моќта (по модул 10^9+7) на почетното дрво. Во i-тата линија од преостанатите q линии ($i \in \{1, \ldots, q\}$), отпечатете ја моќта и супер-моќта (по модул 10^9+7) на дрвото после i-тото ажурирање.

Ограничувања на влезот

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

Бодување

За даден тест случај, твоето решение ќе добие 50% од поените ако ги дава сите точни вредности за моќта, но дава барем една неточна супер-моќ вредност во тој тест случај.

Слично, 50% од поените ќе бидат доделени ако решението ги дава сите точни вредности за супер-моќта, но дава барем една неточна вредност за моќ во тој тест случај.

Подазадачи

1. (4 поени) $n=3$.
2. (7 поени) $n, q \leq 700$.
3. (13 поени) $n, q \leq 5000$.
4. (6 поени) $n \leq 10^5$, $p_i = i-1$ (за секое $i \in \{1, \ldots, n-1\}$), и $a_i, x < 2^{20}$ (за секое $i \in \{0, \ldots, n-1\}$ и за секое ажурирање (v, x)).
5. (7 поени) $p_i = i-1$ (за секое $i \in \{1, \ldots, n-1\}$).
6. (12 поени) $a_i, x < 2^{20}$ (за секое $i \in \{0, \ldots, n-1\}$ и за секое ажурирање (v, x)).
7. (14 поени) $n \leq 10^5$.
8. (11 поени) $n \leq 5 \cdot 10^5$.
9. (26 поени) Без дополнителни ограничувања.

Пример за тест случај 1

Влез

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

Излез

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

Објаснување

На почетокот, имаме

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

Па следи дека, моќта на дрвото е еднаква на:

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

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

Супер-моќта на:

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

По првото ажурирање:

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

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

По второто ажурирање:

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

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

По третото ажурирање:

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

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

Пример за тест случај 2

Влез

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

Излез

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

Објаснување

На почетокот, имаме

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

По првото ажурирање:

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

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

По второто ажурирање:

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

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

Пример за тест случај 3

Влез

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

Излез

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