Super Tree

Vi se oferă un arbore cu rădăcină cu n vârfuri, identificați prin indicii 0, $\ldots$, $n-1$. Rădăcina are indicele 0. Pentru fiecare $i \in \{0, \ldots, n-1\}$, vârful i (vârful cu indicele i) are un număr întreg a_i asociat lui. Fie f_v valoarea ȘI-ului pe biți (notat mai departe cu $\&$) al valorilor a_i pe drumul simplu de la vârful v către rădăcină. (Observați că drumul simplu de la un vârf x la un vârf y conține și vârfurile x și y). Fie *puterea* unui arbore valoarea

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

și fie *superputerea* unui arbore valoarea (atenție la diferența combinațiilor însumate)

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

Pentru un exemplu clarificator, citiți explicațiile exemplelor de mai jos.

Spunem că un vârf u aparține *subarborelui vârfului* v daca v aparține drumului simplu de la vârful u către rădăcină. Observați că subarborele vârfului x conține și vârful x.

Vi se dau q actualizări. Fiecare actualizare este descrisă prin două numere întregi, v și x, și vă cere să setați $a_u := a_u \mathbin{\&} x$ pentru fiecare vârf u din subarborele vârfului v. După fiecare actualizare trebuie să afișați puterea și superputerea arborelui actual.

Deoarece valorile pot fi foarte mari, afișați-le modulo $10^9 + 7$.

Format intrare

Prima linie de la intrare va conține numerele întregi n și q.

A doua linie de la intrare va conține $n-1$ numere întregi, mai exact p_1, p_2, $\ldots$, p_{n-1}, care reprezintă structura arborelui. Pentru fiecare $i \in \{1, \ldots, n-1\}$, p_i este indicele părintelui vârfului i și îndeplinește condiția $0 \le p_i < i$.

A treia linie de la intrare va conține n numere întregi, a_0, a_1, $\ldots$, a_{n-1}. Acestea sunt valorile asociate vârfurilor.

Urmatoarele q linii vor conține câte două numere întregi: v ($0 \le v < n$) și x. Aceste numere descriu actualizările individuale.

Format ieșire

Afișați $q+1$ linii. Fiecare linie trebuie să conține două numere întregi separate printr-un spațiu. Pe prime linie afișați puterea și superputerea (modulo 10^9+7) al arborelui inițial. Pe cea de-a i-a linie din urmatoarele q linii ($i \in \{1,\ldots,q\}$) afișați puterea și superputerea (modulo 10^9+7) al arborelui dupa cea de-a i-a actualizare.

Restricții

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

Punctaj

Pentru fiecare test soluția voastră va primi 50% din punctaj dacă afișează corect toate valorile putere din acel test, dar afișează un răspuns incorect la o valoarea superputere cel puțin o data.

Asemănător, 50% din punctjaul pe acel test va fi acordat soluții voastre daca calculați corect toate valorile superputere din acel test, dar afișați un răspuns incorect la o valoare putere cel puțin o data în acel test.

Subtask-uri

1. (4 puncte) $n=3$.
2. (7 puncte) $n, q \leq 700$.
3. (13 puncte) $n, q \leq 5000$.
4. (6 puncte) $n \leq 10^5$, $p_i = i-1$ (pentru $i \in \{1,\ldots,n-1\}$), și $a_i, x < 2^{20}$ (pentru $i \in \{0,\ldots,n-1\}$ și pentru fiecare actualizare (v,x)).
5. (7 puncte) $p_i = i-1$ (pentru fiecare $i \in \{1,\ldots,n-1\}$).
6. (12 puncte) $a_i, x < 2^{20}$ (pentru fiecare $i \in \{0,\ldots,n-1\}$ și pentru fiecare actualizare (v,x)).
7. (14 puncte) $n \leq 10^5$.
8. (11 puncte) $n \leq 5 \cdot 10^5$.
9. (26 de puncte) Nicio restricție suplimentară.

Exemplu 1

Intrare

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

Ieșire

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

Explicație

Inițial avem

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

Astfel, puterea arborelui este egală cu

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

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

Superputerea este egală cu

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

După prima actualizare:

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

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

După a doua actualizare:

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

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

După a treia actualizare:

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

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

Exemplu 2

Intrare

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

Ieșire

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

Explicație

Inițial avem

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

După prima actualizare:

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

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

După a doua actualizare:

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

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

Exemplu 3

Intrare

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

Ieșire

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