XORanges

Janicko ma rad pomarance! Zostrojil si preto speciany skener pomarancov. Pomocou kamier a Raspberry Pi 3b+, zacal vytvarat 3D snimky pomarancov. Na spracovani signalu este treba co to vylepsit, lebo na vystupe dostava len 32-bitove cele cislo, ktore obsahuje informaciu o skvrnach na supe pomaranca. 32-bitove cele cislo D reprezentuje postupnost 32 cislic (bitov) z ktorych kazda nadobuda hodnotu 0 alebo 1. Ak zacne od 0 mozme skonstruovat D pridanim 2^i pre kazdy nenulovy i-ty bit/cislicu. Formalne D is reprezentuje postupnost $d_{31}, d_{30}, \ldots d_0$ kde $D = d_{31} \cdot 2^{31} + d_{30} \cdot 2^{30} + \ldots + d_1 \cdot 2^1 + d_0 \cdot 2^0$. Napriklad, 13 je reprezentovane ako $0, \ldots, 0, 1, 1, 0, 1$.

Janicko zoskanoval n pomarancov; ale potom sa zrazu rozhodol ze niekedy zoskenuje znova niektory z pomarancov (i-ty) pocas behu Tvojho programu. To znamena ze od tohto okamihu , sa bude pouzivat nova hodnota celeho 32 bitoveho cisla pre pomaranc i.

Janicko chce aj analyzovat svoje pomarance. Na tento ucel sa mu velmi hodi operacia exclusive or (XOR) a preto sa rozhodne urobit pomozou nej zopar vypoctov. Vyberie si postupnost zacinajuc l-tym konciac u-tym (kde $l \leq u$) kde chce pouzit XOR na vsetkych clenoch postupnosti, dalej na vsetkych dvojiciach po sebe iducich cisel z postupnosti, vsetkycf 3 po sebe nasledujucich clenoch postupnosti, ... az po $u - l + 1$ po sebe nasledujucih clenoch (vsetky cleny vo vybranej postupnosti).

teda ak $l = 2$ a $u = 4$ a pole skenerom generovanych hodnot je A, program vypocita hodnotu $a_2 \oplus a_3 \oplus a_4 \oplus (a_2 \oplus a_3) \oplus (a_3 \oplus a_4) \oplus (a_2 \oplus a_3 \oplus a_4)$, kde $\oplus$ reprezentuje XOR a a_i reprezentuje i-ty prvok v poli A.

Definujme binarny XOR operand nasledovne:

Ak i-ty bit prveho cisla/argumentu je rovnaky ako i-ty bit druheho argumentu, potom i-ty bit vysledku je `0`; Ak i-ty bit prveho cisla je rozny od i-teho bitu druheho cisla, potom i-ty bit vysledku je `1`.

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

Napriklad, $13 \oplus 23 = 26$.

$13 =$	$0 \ldots 001101$
$23 =$	$0 \ldots 010111$
$13 \oplus 23 = 26 =$	$0 \ldots 011010$

Vstup

Na prvom vstupnom riadku sa nachadzaju 2 kladne cele cisla n a q (celkovy pocet povelov pre prikaz rescan a analyza vykonanych pocas behu programu).

Na dalsom riadku, je n medzerou oddelenych nezapornych celych cisel, ktore predstavuju hodnoty v poli A (vysledky scanovania pomarancov). Prvok a_i predstavuje hodnotu pre i-ty pomaranc. Index i zacina od 1.

Povely su popisane v nasledujucich q riadkoch, kazdy obsahuje tri medzerou oddelene kladne cisla.

Ak je povel typu 1 (rescan), potom prve cele cislo na vstupnom riadku je 1 nasledovane i (index pomarncu ktory Janicko znova zoskenoval) a j (nova hodnota i-teho pomaranca ktora vysla zo skenera).

Ak je povel typu 2 (analyse), potom prve cislo na riadku je 2 nasledovane indexami l a u.

Vystup

Pre kazdy povel typu 2 (analyse) vypiste hodnotu vysledku na novy riadok, pre povely 1 (scan) netreba vypisovat nic.

Obmedzenia

- $a_i \leq 10^9$
- $0 < n, q \leq 2 \cdot 10^5$

Podulohy

1. **[12 points]**: $0 < n, q \leq 100$
2. **[18 points]**: $0 < n, q \leq 500$ a neobsahuje povel typu 1 (rescan)
3. **[25 points]**: $0 < n, q \leq 5000$
4. **[20 points]**: $0 < n, q \leq 2 \cdot 10^5$ a neobsahuje povel typu 1 (rescan)
5. **[25 points]**: bez obmedzeni

Examples

Priklad 1

Vstup

```
3 3
1 2 3
2 1 3
1 1 3
2 1 3
```

Vystup

```
2
0
```

Poznamky

Na zaciatku, $A = [1, 2, 3]$. Prvy povel je pre cele pole. Vysledok analyzi je $1 \oplus 2 \oplus 3 \oplus (1 \oplus 2) \oplus (2 \oplus 3) \oplus (1 \oplus 2 \oplus 3) = 2$.

Hodnote pre prvy pomaranc sa potom zmeni na 3. To ma za nasledok zmenu odpovede pri povele analyse (opat na pnom poli $[1, 3]$) $3 \oplus 2 \oplus 3 \oplus (3 \oplus 2) \oplus (2 \oplus 3) \oplus (3 \oplus 2 \oplus 3) = 0$.

Priklad 2

Vstup

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

Vystup

```
2
5
4
4
```