

XORтокали

Яниз обожава портокали! Затова направи скенер за портокали. С камера и Raspberry Pi 3b+ компютър, той започна да създава 3D образи на портокали. Неговият графичен процесор не е много добър, затова единствения изход, който получава е едно 32-битово число, което съдържа информация за дупките (вдлъбнатините) на кората. 32-битовото число D представлява редица от 32 цифри (битове), всяка от които е едно или нула. Ако започнем от 0, можем да получим D като добавим 2^i за всеки i -ти бит, който е единица. По формално, числото D се представя от редицата $d_{31}, d_{30}, \dots, d_0$, където $D = d_{31} \cdot 2^{31} + d_{30} \cdot 2^{30} + \dots + d_1 \cdot 2^1 + d_0 \cdot 2^0$. Например, 13 се представя като $0, \dots, 0, 1, 1, 0, 1$.

Яниз сканира n портокала, въпреки че от време на време решава повторно да сканира някой от портокалите по време на изпълнението на вашата програма. Това означава, че от този момент нататък, той използва новата стойност за i -тия портокал.

Яниз иска да анализира тези портокали. Той открива, че операцията изключващо или (XOR) е много интересна, и затова решава да направи някои сметки с нея. Яниз избира поредица от портокали от индекс l до индекс u (където $l \leq u$) и иска да намери стойността на XOR на: всички елементи в тази поредица, на всички двойки от последователно разположени елементи, всички последователности от 3 елемента, ... и единствената поредица от $u - l + 1$ последователно разположени елемента (всички елементи в поредицата).

Т.е. ако $l = 2$ и $u = 4$ и имаме масив от сканирани стойности A , програмата трябва да върне стойността на $a_2 \oplus a_3 \oplus a_4 \oplus (a_2 \oplus a_3) \oplus (a_3 \oplus a_4) \oplus (a_2 \oplus a_3 \oplus a_4)$, където \oplus представлява XOR и a_i представлява i -тия елемент на масива A .

XOR операцията е дефинирана както следва:

Ако i -тия бит на първото число е същият като i -тия бит на второто число, тогава i -тия бит на резултата е 0, а ако i -тият бит на първото число е различен от i -тия бит на второто число, i -тият бит на резултата е 1.

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

Например, $13 \oplus 23 = 26$.

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

Вход

На първия ред на стандартния вход са 2 положителни цели числа n (броят на портокалите) и q (общият брой на новите сканирания и операциите по изследване).

На следващия ред, има n неотрицателни цели числа, разделени с интервал, които представят стойностите на масива A (резултатите от сканиране на портокалите). Елементът a_i съдържа стойността на i -тия портокал. Индексацията е от 1.

Операциите се представени на следващите q реда с 3 положителни цели числа, разделени с интервал.

Ако операцията е от тип 1 (повторно сканиране), първото число е 1 и е последвано от i (индекс на портокал, който Яниз иска да сканира наново) и j (резултата от повторното сканиране на i -тия портокал).

Ако операцията е от тип 2 (изследване), първото число е 2 и е последвано от индексите l и u .

Изход

За всяка операция от тип 2 (изследване) трябва да отпечатате едно-единствено число — стойността за съответната заявка. Трябва да отпечатате всяко число на нов ред по реда на въвеждане на операциите за изследване.

Ограничения

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

Подзадачи

1. **[12 точки]**: $0 < n, q \leq 100$
2. **[18 точки]**: $0 < n, q \leq 500$ и няма повторни сканирания (операция от тип 1)
3. **[25 точки]**: $0 < n, q \leq 5000$
4. **[20 точки]**: $0 < n, q \leq 2 \cdot 10^5$ и няма повторни сканирания (операция от тип 1)
5. **[25 точки]**: Няма допълнителни ограничения.

Примери

Пример 1

Вход

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

Изход

```
2
0
```

Обяснение

В началото, $A = [1, 2, 3]$. Първата операция за изследване е за цялата поредица. Резултата от това е: $1 \oplus 2 \oplus 3 \oplus (1 \oplus 2) \oplus (2 \oplus 3) \oplus (1 \oplus 2 \oplus 3) = 2$.

Тогава стойността на първия портокал се променя на 3. Това води до промяна на същото изследване (за същия интервал $[1, 3]$): $3 \oplus 2 \oplus 3 \oplus (1 \oplus 2) \oplus (2 \oplus 3) \oplus (1 \oplus 2 \oplus 3) = 0$.

Пример 2

Вход

```
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
```

Изход

2
5
4
4