Problem Fiboxor

| C header | fiboxor.h |
| C++ header | fiboxor.h |

Little Square and Little Triangle are having a fight on Valentine's day again! The reason of this year's fight? Little Square has said that the Fibonacci sequence is way prettier than the sequence of powers of two (Little Triangle's favorite). Recall that the Fibonacci sequence is the sequence that begins with 0 and 1, and where subsequent elements are the sum of the two previous elements:

$$0, 1, 1, 2, 3, 5, 8, 13, \ldots$$

Little Triangle likes this sequence particularly because you can easily calculate the bitwise exclusive or (XOR) of any subarray. Thus they told Little Square that if they can successfully answer Q such XOR subarray queries, then they will accept the fact that Fibonacci is better. Little Triangle isn't cruel though, so they agreed to let Little Square tell them only the value of the subarray XOR modulo 2^k each query, due to the fact that the Fibonacci sequence grows exponentially.

In the end, Little Square has to answer Q queries, each of which contains three integers k, l, r. This query asks us to calculate the XOR of the Fibonacci numbers with indexes in the interval $[l, r]$ inclusive, indexed from 0, modulo 2^k.

Little Square asks for your help in order to win this argument, so they can go back at having a good time on Valentine's day.

Interaction protocol

The contestant must implement two functions, `init` and `query`, with the following signature.

```
void init(int q);
int query(int k, long long l, long long r);
```

The first function will be called by the committee exactly once, before any calls to `query`, and will be supplied the value of Q. The second function must answer the query determined by integers k, l, r, and return the answer. It will be called at most Q times by the grader.

Constraints

- $1 \leq Q \leq 10^6$
- $0 \leq l \leq r \leq 10^{18}$
- $1 \leq k \leq 20$

Subtask 1 (9 points)

- $k = 1$

Subtask 2 (18 points)

- $Q \leq 10^5$
- $r - l + 1 \leq 10$

Subtask 3 (30 points)

- $r \leq 10^6$

Subtask 4 (43 points)

- No additional constraints.

Example

input	output
3	2
2 0 3	14
4 4 6	519891
20 100 1000	

Explanation

Let $a \oplus b$ denote the XOR of a and b.

In the first query, $0 \oplus 1 \oplus 1 \oplus 2 \mod 2^2 = 2 \mod 4 = 2$.

In the second query, $3 \oplus 5 \oplus 8 \mod 2^4 = 14 \mod 16 = 14$.