E. Binary Wine

time limit per test: 2 seconds
memory limit per test: 512 megabytes

You are given n integers $a_1, a_2, \ldots, a_n$ within the range $[0, 2^{30})$.

You can spend 1 coin to increase any a_i by 1. You can perform this operation any number of times.

You need to solve q queries; for each query, you are given an integer c, also in the range $[0, 2^{30})$. You would like it if there exists a sequence b of length n with the following properties:

- For every $1 \leq i \leq n$, $0 \leq b_i \leq a_i$.
- $b_1 \oplus b_2 \oplus \ldots \oplus b_n = c$, where $\oplus$ denotes the bitwise XOR operation.

Please calculate the minimum number of coins you will have to spend, such that there exists a suitable b.

The queries are **independent**, meaning that any operations you perform on the sequence a will not impact future queries.

Codeforces Round 1064 (Div. 2)
比赛进行中
01:02:30
Contestant

→ 提交?

语言: GNU G++20 13.2 (64 bit, w

选择文件: 选择文件 未选择文件

Be careful: there is 50 points penalty for submission which fails the pretests or resubmission (except failure on the first test, denial of judgement or similar verdicts). "Passed pretests" submission verdict doesn't guarantee that the solution is absolutely correct and it will pass system tests.

提交

→ 评分表

	Score
Problem A	386

您可以花费 1 枚硬币，将任意 a_i 增加 1 。您可以多次执行此操作。

您需要解决 q 个查询；对于每个查询，您都会得到一个同样在 $[0, 2^{30})$ 范围内的整数 c 。如果存在长度为 n 的序列 b 且具有以下属性，您会希望它存在：

- 对于每个 $1 \le i \le n$ ， $0 \le b_i \le a_i$ 。
- $b_1 \oplus b_2 \oplus \ldots \oplus b_n = c$ ，其中 $\oplus$ 表示 **bitwise XOR 运算**。

请计算你需要花费的最小金币数，以便存在一个合适的 b 。

查询是**独立的**，这意味着您对序列 a 执行的任何操作都不会影响以后的查询。

Problem B	579
Problem C	772
Problem D	1158
Problem E	1544
Problem F	2316
Successful hack	100
Unsuccessful hack	-50
Unsuccessful submission	-50
Resubmission	-50

* If you solve problem on 00:57 from the first attempt

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case consists of two integers n, q ($1 \le n \le 5 \cdot 10^5$, $1 \le q \le 5 \cdot 10^4$) — the length of sequence a and the number of queries.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i < 2^{30}$) — the initial sequence a.

Each of the next q lines contains a single integer c ($0 \le c < 2^{30}$) — the target XOR.

It is guaranteed that the sum of n over all test cases does not exceed $5 \cdot 10^5$.

It is guaranteed that the sum of q over all test cases does not exceed $5 \cdot 10^4$.

DeepL 翻译

输入

每个测试包含多个测试用例。第一行包含测试用例的数量 t （ $1 \le t \le 10^4$ ）。测试用例说明如下。

每个测试用例的第一行包含两个整数 n, q （ $1 \le n \le 5 \cdot 10^5$ ， $1 \le q \le 5 \cdot 10^4$)--序列长度 a 和查询次数。

每个测试用例的第二行包含 n 个整数 $a_1, a_2, \ldots, a_n$ （ $0 \le a_i < 2^{30}$ ）- 初始序列 a 。

接下来的每行 q 都包含一个整数 c （ $0 \le c < 2^{30}$ ）- 目标 XOR。

保证所有测试用例中 n 的总和不超过 $5 \cdot 10^5$ 。

保证所有测试用例中 q 的总和不超过 $5 \cdot 10^4$ 。

Output

For each query, output a single integer — the minimum coins you will have to spend, such that there exists a suitable b.

DeepL 翻译

输出

对于每个查询，输出一个整数--在存在合适的 b 的情况下，您需要花费的最小硬币数。

Example

input Copy

```
4
2 1
5 7
9
3 1
9 9 8
24
6 4
1 1 4 5 1 4
10
20
30
40
1 1
0
0
```

output Copy

```
1
7
3
11
16
31
```

0

Note

In the first test case, we spend 1 coin to increase a_2 by 1, resulting in sequence $[5, 8]$. A suitable b would be $[1, 8]$. It can be shown one cannot spend less than 1 coin to achieve the objective.

In the second test case, we can spend 7 coins to increase a_1 by 7, resulting in sequence $[16, 9, 8]$. A suitable b would be $[16, 9, 1]$.

DeepL 翻译

注

在第一个测试案例中，我们花费 1 个硬币将 a_2 增加 1 ，结果序列为 $[5, 8]$ 。合适的 b 是 $[1, 8]$ 。可以证明，要达到目标，花费的硬币不能少于 1 个。

在第二个测试案例中，我们可以花费 7 枚硬币将 a_1 增加 7 ，得到序列 $[16, 9, 8]$ 。合适的 b 是 $[16, 9, 1]$ 。

GNU G++20 13.2 (64 bit, winlibs)

```
1
```

▶ 自定义测试数据(自动保存)