



Minimum xor

locked

Problem

Submissions

Leaderboard

Discussions

В задачата ще трябва да отговорите на Q на брой заявки върху множество от числа S .

Първоначално множеството S съдържа само 1 елемент - 0 ($S = \{0\}$). При всяка заявка се въвежда едно цяло число P_i , което се добавя към множеството (S не е мултимножество \implies ако числото P_i вече се среща в множеството, то не трябва да бъде добавено втори път).

От вас се иска след всяка заявка да изведете по едно цяло число - минималната стойност която може да се получи чрез прилагане на **xor** (побитово изключващо или: \oplus) на някои 2 елемента принадлежащи на множеството.

По формално казано, след всяка заявка намерете: $\min(\{u \oplus v | \{u, v\} \subseteq S\})$.

Input Format

Първият ред на стандартния вход съдържа едно цяло число Q - броя на заявките.

Следват Q на брой цели числа P_i - поредното число което трябва да бъде добавено в множеството S .

Constraints

$$0 \leq Q \leq 10^5$$

$$1 \leq P_i \leq 10^9$$

Output Format

Изведете Q на брой реда с по едно цяло число на всеки ред - търсената стойност за всяка от заявките.

Sample Input 0

```
5
7
3
5
5
42
```

Sample Output 0

```
7
3
2
2
2
```

Explanation 0

$$S = \{0\} \cup \{7\} = \{0, 7\} \implies \text{минималният xor е: } (0 \oplus 7) = 7.$$

$$S = \{0, 7\} \cup \{3\} = \{0, 7, 3\} \implies \text{минималният xor е } (0 \oplus 3) = 3.$$

$$S = \{0, 7, 3\} \cup \{5\} = \{0, 7, 3, 5\} \implies \text{минималният xor е: } (7 \oplus 5) = 2.$$

$$S = \{0, 7, 3, 5\} \cup \{5\} = \{0, 7, 3, 5\} \implies \text{минималният xor е: } (7 \oplus 5) = 2.$$



$$S = \{0, 7, 3, 5\} \cup \{42\} = \{0, 7, 3, 5, 42\} \implies \text{минималният xor е: } (7 \oplus 5) = 2.$$

Submissions: 83

Max Score: 100

Difficulty: Hard

Rate This Challenge:

[More](#)Current Buffer (saved locally, editable)  

C++14



```
1 #include <cmath>
2 #include <cstdio>
3 #include <vector>
4 #include <iostream>
5 #include <algorithm>
6 using namespace std;
7
8
9 int main() {
10     /* Enter your code here. Read input from STDIN. Print output to STDOUT */
11     return 0;
12 }
```

Line: 1 Col: 1

 [Upload Code as File](#) ☐ [Test against custom input](#)[Run Code](#)[Submit Code](#)