

And Operation TAAND

Given an array of n non-negative integers: A_1, A_2, \dots, A_N . Your mission is finding a pair of integers A_u, A_v ($1 \leq u < v \leq N$) such that $(A_u \text{ bit_and } A_v)$ is as large as possible.
(**And** is a bit-wise operation which is corresponding to `&` in C++)

Input

The first line of the input contains a single integer N . The i -th line in the next N lines contains the A_i .

Output

Contains a single integer which is the largest value of A_u and A_v where $1 \leq u < v \leq N$.

Constraints

50 points:

- $2 \leq N \leq 5000$
- $0 \leq A_i \leq 10^9$

50 points:

- $2 \leq N \leq 3 \times 10^5$
- $0 \leq A_i \leq 10^9$

Example

Input:

4
2
4
8
10

Output:

8

Explanation

- 2 and 4 = 0
- 2 and 8 = 0
- 2 and 10 = 2
- 4 and 8 = 0
- 4 and 10 = 0
- 8 and 10 = 8