Swapping Operation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Given a non-negative integer sequence $A = a_1, a_2, \ldots, a_n$ of length n, define

$$F(A) = \max_{1 \le k < n} \left((a_1 \,\&\, a_2 \,\&\, \cdots \,\&\, a_k) + (a_{k+1} \,\&\, a_{k+2} \,\&\, \cdots \,\&\, a_n) \right)$$

where & is the bitwise-and operator.

You can perform the swapping operation at most once: choose two indices i and j such that $1 \le i < j \le n$ and then swap the values of a_i and a_j.

Calculate the maximum possible value of $F(A)$ after performing at most one swapping operation.

Input

There are multiple test cases. The first line of the input contains an integer T indicating the number of test cases. For each test case:

The first line contains an integer n ($2 \le n \le 10^5$) indicating the length of sequence A.

The second line contains n integers $a_1, a_2, \cdots, a_n$ ($0 \le a_i \le 10^9$) indicating the given sequence A.

It's guaranteed that the sum of n of all test cases will not exceed 10^5.

Output

For each test case output one line containing one integer indicating the maximum possible value of $F(A)$ after performing at most one swapping operation.

Example

standard input	standard output

3
6
6 5 4 3 5 6
6
1 2 1 1 2 2
5
1 1 2 2 2
	7
	3
	3

Note

For the first sample test case, we can swap a_4 and a_6 so the sequence becomes $\{6, 5, 4, 6, 5, 3\}$. We can then choose $k = 5$ so that $F(A) = (6 \,\&\, 5 \,\&\, 4 \,\&\, 6 \,\&\, 5) + (3) = 7$.

For the second sample test case, we can swap a_2 and a_4 so the sequence becomes $\{1, 1, 1, 2, 2, 2\}$. We can then choose $k = 3$ so that $F(A) = (1 \,\&\, 1 \,\&\, 1) + (2 \,\&\, 2 \,\&\, 2) = 3$.

For the third sample test case we do not perform the swapping operation. We can then choose $k = 2$ so that $F(A) = (1 \,\&\, 1) + (2 \,\&\, 2 \,\&\, 2) = 3$.