Chang and Bitwise OR Problem Code: **CHNGOR**

Chang is a big fan of bitwise operations. As he was sitting in a boring lecture, he thought of a simple problem but had a tough time figuring out the solution. Help him to solve it. The problem is as follows.

Given a list of **N** non-negative integers, you perform the following operation on this list.

- Select any subsequence of integers from the list and remove the elements of that subsequence. The cost incurred will be Bitwise OR of the elements.

Your task is to remove all the integers from the list by applying the above operation as many times as you want. You want to incur the minimum total cost at the end. Total cost is the sum of all the costs incurred while removing subsequences.

Input

The first line of the input contains an integer **T** denoting the number of test cases.

The first line of the each test case contains a single integer **N** denoting the number of integers in the list.

The second line contains **N** space-separated integers A_1, A_2, ..., A_N denoting the integers in the given list.

Output

For each test case, output answer in a separate line.

Constraints

- $1 \leq T \leq 10$
- $1 \leq N \leq 10^5$
- $0 \leq A_i \leq 10^9$

Example

```
Input:

1

2

1 2



Output:

3
```

Explanation

Example case 1.

We can first remove **1** from the list and then **2**. Thus, the total cost comes out to **3**.

The other possible way is to choose the subsequence **1**, **2**, and remove it. The Bitwise OR of **1** and **2** is **3**. So the cost incurred will be **3** in this case too.