**Find first set bit**

Show Topic Tags   

[Amazon](http://practice.geeksforgeeks.org/company/Amazon/)

Given an integer an N, write a program to print the position of first set bit found from right side in the binary representation of the number.

**Input:**  
The first line of the input contains an integer T, denoting the number of test cases. Then T test cases follow. The only line of the each test case contains an integer N.  
  
**Output:**  
For each test case print in a single line an integer denoting the position of the first set bit found form right side of the binary representation of the number.  
  
**Constraints:**  
1<=T<=100  
0<=N<=106  
  
**Example:**  
**Input:**  
2  
18  
12

**Output:**  
2  
3

**Explanation:**  
Test case 1:  
Binary representation of the 18 is 010010, the first set bit from the right side is at position 2.

\*\*For More Examples Use Expected Output\*\*

<http://practice.geeksforgeeks.org/problems/find-first-set-bit/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication241;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.HashSet;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication241 {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) throws IOException {

// TODO code application logic here

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(br.readLine());

while(t-- > 0) {

int n = Integer.parseInt(br.readLine());

if(n ==0) {

System.out.println(0);

continue;

}

String bin = Integer.toString(n,2);

int pos =1;

for(int i = bin.length()-1; i>=0; i--) {

if(bin.charAt(i ) == '1') {

break;

}

pos++;

}

System.out.println(pos);

}

}

}