**Find first set bit: -**

Easy Accuracy: 46.89% Submissions: 130K+ Points: 2

Given an integer **N.** The task is to return the position of first set bit found from the **right**side in the binary representation of the number.  
**Note**: If there is no set bit in the integer N, then return 0 from the function.

**Example 1:**

**Input**:   
N = 18

**Output**:   
2

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

**Example 2:**

**Input**:   
N = 12

**Output**:   
3

**Explanation**:   
Binary representation of 12 is 1100, the first set bit from the right side is at position 3.

**Your Task:**  
The task is to complete the function **getFirstSetBit**() that takes an integer **n** as a parameter and returnstheposition of first set bit.

**Expected Time Complexity:** O(log N).  
**Expected Auxiliary Space:** O(1).

**Constraints:**  
0 <= N <= 108

**Code: -**

//{ Driver Code Starts

//Initial Template for C++

#include<bits/stdc++.h>

using namespace std;

// } Driver Code Ends

//User function Template for C++

class Solution

{

public:

//Function to find position of first set bit in the given number.

unsigned int getFirstSetBit(int n){

int count = -1;

while(++count < 32){

if((1 << count) & n)

return count+1;

}

return 0;

}

};

//{ Driver Code Starts.

// Driver code

int main()

{

int t;

cin>>t; // testcases

while(t--)

{

int n;

cin>>n; //input n

Solution ob;

printf("%u\n", ob.getFirstSetBit(n)); // function to get answer

}

return 0;

}

// } Driver Code Ends

**T.C: - O(1), max iterating for 32(bits)**

**S.C: - O(1)**