Minimum Step To One

Problem Statement:

Problem statement is very easy . On a positive integer, you can perform any one of the following 3 steps.

1.) Subtract 1 from it. (n = n - 1)

2.) If its divisible by 2, divide by 2. (if n % 2 == 0 , then n = n / 2)

3.) If its divisible by 3, divide by 3. (if n % 3 == 0 , then n = n / 3)

Given a positive integer n and you task is find the minimum number of steps that takes n to one .

Input:

The input contains an integer **T** (1 ≤ **T** ≤ 100) number of test cases. Second line input is N (0 < **N** ≤ $2*10^{7}$) that indicates the positive number.

Output:

For each case, print the case number and minimum steps.

Sample Input/Output:

Sample Input	Sample Output
3 **1** **4** **7**	**Case 1: 0** **Case 2: 2** **Case 3: 3**

For example :-

1.) For N= 1 , output: 0

2.) For N = 4 , output: 2 (4 /**2** = 2 /**2** = 1)

3.) For N = 7 , output: 3 (7 **-1** = 6 /**3** = 2 /**2** = 1)