1032 – Fast Bit Calculations

A bit is a binary digit, taking a logical value of either **1** or **0** (also referred to as "true" or "false" respectively). And every decimal number has a binary representation which is actually a series of bits. If a bit of a number is **1** and its next bit is also **1** then we can say that the number has a **1** adjacent bit. And you have to find out how many times this scenario occurs for all numbers up to **N**.

Examples:

Number	Binary	Adjacent Bits
12	1100	1
15	1111	3
27	11011	2

Input

Input starts with an integer **T (≤ 10000)**, denoting the number of test cases.

Each case contains an integer **N (0 ≤ N < 2^{31})**.

Output

For each test case, print the case number and the summation of all adjacent bits from **0** to **N**.

Sample Input	Output for Sample Input
7 0 6 15 20 21 22 2147483647	Case 1: 0 Case 2: 2 Case 3: 12 Case 4: 13 Case 5: 13 Case 6: 14 Case 7: 16106127360