Buying Integers

Let's assume that you have **n** integers, $A_1, A_2, A_3 \ldots A_n$.

Let's define:

- **E** = Number of pairs **(i, j)** such that **i < j** and **($A_i + A_j$)** are even.
- **O** = Number of pairs **(i, j)** such that **i < j** and **($A_i + A_j$)** are odd.
- **D = | E-O |** (That means, **D = (E-O)** if **(E-O) ≥ 0, -(E-O)** otherwise.)

Unfortunately, you do have **n** but those **n** integers are lost. You will have to buy them again. Before going to the market, you have decided that you will buy **n** integers in such a way that the value of **D** will be as small as possible, as you will have to pay **D** golden coins to buy them.

Now, you are wondering, what that minimum **D** will be. (Let's call it D_{min}).

Input

First line of the input file will contain the number of test cases, **T ≤ 1000000**, followed by **T** lines, each containing an integer **n** ($1 \leq n \leq 10^9$).

Output

For each case, print the case number starting from 1 and D_{min} for the value of **n** in that particular case. See the sample output for exact formatting.

Example

Input:
3
3
4
5

Output:
Case 1: 1
Case 2: 0
Case 3: 2

Warning: Input file is huge, please use faster input and output methods (e.g. printf and scanf in C++).

Problem Setter: Momontho Mashak Monmoy

Special Thanks: Muhammad Ridowan