**Flipper-McFlopper**

*Filename:* flipper

The star-bellied Sneetches, they want to have fun

Blooping and flooping till Sunday is done.

But one little chore they have left to do:

Test their Flipper-McFlopper machine for dear Sue.

The Flipper-McFlopper machine is quite grand. It takes in a single, quite large, operand.

We'll call it *x*, and so that you know, what happens inside is recorded below:

It does this until *x* ends in a 7 -- first add 1 or 2, then scale by 11.

The Flipper-McFlopper should always choose best to stop extra fast for every new test.

Find the fewest number of scales to get to the end,

assuming optimal picks (each either 1 or 2) for every addend.

**The Problem:**

Given x, calculate the minimum number of scales the Flipper-McFlopper must do before stopping.

**The Input:**

The first line of the input file begins with a single, positive integer, *t*, representing the number of test cases. *t* lines follow, each containing a single integer, 1 ≤ x ≤ 1018, the input to the machine.

**The Output:**

For each test case, output a single line saying “Input #i: s” without the quotes, where i is the number of the test case, and s the minimum number of scales required.

**Sample Input:**

3

197

4

583920682924895290

**Sample Output:**

Input #1: 0

Input #2: 2

Input #3: 4

**Note:** For the first case the Flipper-McFlopper machine will stop instantly since the number already ends in a 7. For the second test case, ((4 +1)\*11 +2)\*11 = 627. It will stop here after multiplying twice. It is also possible to stop at 737 after 2 scales.