Huge Power

Input file:	`math.in`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Hala loves making monsters in her house garden..!!. Each day, she makes a new one. The monster who was made on the i_{th} day has a_i *power*.

Hala noticed that each new monster has more power than all the old monsters' powers combined. After hard calculations, she concluded that all monsters give *their powers* to the new monster and Hala supports the new monster by *double of the average* of the old monsters' powers, mathematically:

$$a_n = a_{n-1} + a_{n-2} + \ldots + a_1 + 2 \times \frac{a_{n-1} + a_{n-2} + \ldots + a_1}{n-1}$$

Hala is in the middle of the fight with her Mom, so she asks you to determine the sum of the first n monsters' powers modulo $10^9 + 7$.

(We assume that the first monster has 1 power; $a_1 = 1$).

Input

The first line contains a single integer T ($1 \leq T \leq 10^5$), the number of test cases.

Each test case contains a single integer n ($1 \leq n \leq 10^{10}$).

Output

For each test case, print a single integer, the sum of the first n monsters' powers modulo $10^9 + 7$.

Example

math.in	standard output
2 2 5	4 80

Note

For the first test case: the first monster has 1 *power* the second has $1 + 2(\frac{1}{1}) = 3$, the total is 4