Fibonacci Power Sum

Some people may found FIBOSUM a too easy problem. We propose here some serious constraints.

$$\sum_{i=1}^{N} \mathrm{Fib}(i)^k$$

Fib is the Fibonacci sequence:
For any positive integer i: if i<2 Fib(i) = i, else Fib(i) = Fib(i-1) + Fib(i-2)

Input

The input contains several lines, you don't have to process all, it may be hard, only all first ones you are able to, in the given time. Each line contains one integer : ***N***.

Output

For the k^{th} line, print **Sum(Fib(*i*)^*k* for *i* in [1..*N*])**.
As the answer could not fit in a 64-bit container, just output your answer modulo 1000000007.
The difficulty should increase as you will process lines.

Example

Input:
6
6
6
[...] some more lines

Output:
20
104
674
[...] as many as you can.

Explanations

For the first line : 1^1 + 1^1 + 2^1 + 3^1 + 5^1 + 8^1 = 20.
For the second line : 1^2 + 1^2 + 2^2 + 3^2 + 5^2 + 8^2 = 104
For the third line : 1^3 + 1^3 + 2^3 + 3^3 + 5^3 + 8^3 = 674.

Constraints

0 < N <= 10^9

The numbers ***N*** are uniform randomly chosen.
The challenge is to be the fastest. There were 30000 lines at time of publication.

Robert Gerbicz agreed to extend to 100000 lines using his fast C code. Congratulations again to him as a fabulous solver.

Now you have one point for the ***k***th line. Constraints allow everybody to get some points.
If a much faster method is discovered, judge would take time into account.
For your information, my (our) method have an amortized complexity of O(k) for the kth line. **Have fun ;-)**