

# Balanced Tree

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1.5 seconds  
Memory limit:         64 megabytes

A binary tree  $T$  is called **super balanced** if  $T$  is empty or satisfies the following three conditions at the same time:

1. The left subtree of  $T$  is super balanced.
2. The right subtree of  $T$  is super balanced.
3. The number of nodes of the left subtree and the right subtree differs at most 1.

Please calculate the number of **super balanced** trees consisting of  $n$  nodes. Since the answer can be very large, just output the answer modulo  $2^{64}$ .

## Input

The first line contains a single integer  $T$  ( $1 \leq T \leq 10^6$ ), indicating the number of test cases.

Each test case contains a single integer  $n$  ( $0 \leq n < 2^{64}$ ), indicating the number of nodes in the tree.

## Output

For each test case, output an integer in a single line indicating the number of super balanced trees consisting of  $n$  nodes.

## Example

standard input	standard output
2	2
2	1
3	