**Nth Fibonacci Number**

[maths](http://www.practice.geeksforgeeks.org/tag-page.php?tag=maths&isCmp=0)

Given a positive integer N, find the Nth fibonacci number. Since the answer can be very large, print the answer modulo 1000000007.  
  
**Input:**  
  
The first line of input contains T denoting the number of testcases.Then each of the T lines contains a single positive integer N.  
  
**Output:**  
  
Output the Nth fibonacci number.  
  
**Constraints:**  
  
1<=T<=100  
1<=N<=1000  
  
**Example:**  
  
Input:  
3  
1  
2  
5

Output:  
0  
1  
3

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=294>

t = int(raw\_input())

for i in range(0, t):

n= int(raw\_input())

fib = []

fib.append(0)

fib.append(1)

for i in range(2, n + 1):

fib.append(fib[i-2] + fib[i-1])

#print fib[n] % 1000000007

*'''*

*for i in range(0, len(fib)):*

*print i + 1, " " , fib[i]*

*'''*

print fib[n - 1] % 1000000007

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication3;

import java.math.BigInteger;

import java.util.Scanner;

/\*\*

\*

\* @author Usuario

\*/

public class JavaApplication3 {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

// TODO code application logic here

BigInteger[] fib = new BigInteger[1001];

fib[0] = BigInteger.ZERO;

fib[1] = BigInteger.ONE;

for(int i =2; i<1001; i++)

{

fib[i] = fib[i - 1].add( fib[i - 2]);

fib[i] = fib[i].mod( BigInteger.valueOf(1000000007));

}

Scanner sc = new Scanner(System.in);

int t = Integer.parseInt(sc.nextLine().trim());

while(t-- > 0)

{

int n = Integer.parseInt(sc.nextLine().trim());

System.out.println(fib[n]);

}

}

}