**Padovan Sequence**

Show Topic Tags   

A **Padovan Sequence** is a sequence which is represented by the following recurrence  
P(n) = P(n-2) + P(n-3)  
P(0) = P(1) = P(2) = 1  
Now given a number N your task is to find the Nth no in this sequence.  
  
**Note:** Since the output could be very long take mod 1000000007

**Input:**  
The first line of input contains an integer T denoting the no of test cases. Then T test cases follow. Each line contains an integer N.  
  
**Output:**  
For each test case in a new line print the nth no of the Padovan sequence.  
  
**Constraints:**  
1<=T<=100  
1<=N<=100  
  
**Example:  
Input:**  
2  
12  
4  
**Output:**  
21  
2

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

<http://practice.geeksforgeeks.org/problems/padovan-sequence/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication243;

import java.io.\*;

import java.util.ArrayList;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication243 {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) throws IOException {

// TODO code application logic here

Long[] pad = new Long[101];

pad[0]= 1L;

pad[1] = 1L;

pad[2] = 1L;

for(int i =3; i< 101; i++) {

pad[i] = pad[i-2] + pad[i-3];

}

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(br.readLine());

while(t-- > 0) {

int n = Integer.parseInt(br.readLine());

System.out.println(pad[n]% 1000000007);

}

}

}