|  |  |
| --- | --- |
| **Summing Subsets**    Problem code: RESN05 | * [SUBMIT](https://www.codechef.com/submit/RESN05) * [MY SUBMISSIONS](https://www.codechef.com/status/RESN05,nacho0monllor) * [ALL SUBMISSIONS](https://www.codechef.com/status/RESN05) |

**All submissions for this problem are available.**

Let G(S) denote the sum of the elements of set S and F(n) be the sum of G(s) for all subsets of the set consisting of the first n natural numbers. For example, F(3) = (1) + (2) + (3) + (1 + 2) + (1 + 3) + (2 + 3) + (1 + 2 + 3) = 24. Given n, calculate F(1) + F(2) + ... + F(n).

**Input**

The first line contains the number of test cases T (<= 1000). Each of the next T lines contains an integer n. (1 <= n <= 1000000000).

**Output**

Output T lines, one corresponding to each test case. Since the answers can get very big, output the answer modulo 8388608

**Example**

**Input:**

3

1

2

3

**Output:**

1

7

31

<https://www.codechef.com/problems/RESN05>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

int t = int.Parse(Console.ReadLine());

while (t-- > 0)

{

int[] ans = { 1, 7, 31, 111, 351, 1023, 2815, 7423, 18943, 47103, 114687, 274431, 647167, 1507327, 3473407, 7929855, 1179647, 6815743, 6291455, 7340031, 6291455, 8388607 };

//Console.WriteLine(ans.Length);

long n = long.Parse(Console.ReadLine());

if (n > 22) n = 22;

Console.WriteLine(ans[n - 1]);

}

Console.ReadLine();

}

}

}

//para obtener las sumas

<https://www.codechef.com/viewsolution/454460>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.IO;

namespace ConsoleApplication1

{

class Program

{

static List<List<long>> Powerset(long[] nums)

{

List<List<long>> ps = new List<List<long>>();

ps.Add(new List<long>()); // add the empty set

// for every item in the original list

foreach (long item in nums)

{

List<List<long>> newPs = new List<List<long>>();

foreach (List<long> subset in ps)

{

// copy all of the current powerset's subsets

newPs.Add(subset);

// plus the subsets appended with the current item

List<long> newSubset = new List<long>(subset);

newSubset.Add(item);

newPs.Add(newSubset);

}

// powerset is now powerset of list.subList(0, list.indexOf(item)+1)

ps = newPs;

}

return ps;

}

static long sumOfSubsets(long n)

{

List<long> nums = new List<long>();

for (long i = 1; i <= n; i++)

{

nums.Add(i);

}

List<List<long>> ps = Powerset(nums.ToArray());

long sum = 0;

foreach (List<long> lista in ps)

{

foreach (long elem in lista)

{

sum += elem;

}

}

return sum;

}

static void Main(string[] args)

{

//for (long i = 1; i <= 20; i++)

//{

// Console.WriteLine(i+ " " + sumOfSubsets(i));

//}

using (StreamWriter es = new StreamWriter("C:\\ans.txt"))

{

long sum = 0;

for (long n = 1; sum % 8388608 != 8388607; n++)

{

sum = 0;

for (long i = 1; i <= n; i++)

{

sum += sumOfSubsets(i);

}

// Console.WriteLine(sum % 8388608);

es.Write((sum % 8388608) + ", ");

}

}

Console.ReadLine();

}

}

}