**Little Shino and Number of Divisors**

Max. Marks: 100

The challenge is over and this problem has been moved to practice area. You can either submit your solution here or   
[Go to Practice Area](https://www.hackerearth.com/es/problem/algorithm/number-of-divisors-3-4f073391/). Also further submissions won't affect the leaderboard.

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

You are given an integer array AA of size xx denoting the prime powers of an integer NN. AiAi denotes the power of ithith prime in the prime factorization of NN. To make it more clear, A1A1 will denote the power of 22 in the prime factorization of NN, A2A2 will denote the power of 33 in the prime factorization of NN and so on.

Consider a number PP equals to the product of all the divisors of NN. You have to find the number of divisors of PP. Output it modulo 109+7109+7.



**Input Format:**  
The first line contains an integer, xx (1≤x≤1061≤x≤106) denoting the size of array AA. Next line contains xxspace separated integers, denoting the array AA (0≤Ai≤1090≤Ai≤109).

**Output Format:**  
Print one integer, denoting the number of divisors of PP, modulo 109+7109+7.

**SAMPLE INPUT**

3

1 1 1

**SAMPLE OUTPUT**

125

**Explanation**

N=21∗31∗51=30N=21∗31∗51=30

P=1∗2∗3∗5∗6∗10∗15∗30=810000P=1∗2∗3∗5∗6∗10∗15∗30=810000

Number of factors of PP is 125125

**Time Limit:**1,0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded if any testcase passes.

**Allowed Languages:**C, C++, C++14, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift, Visual Basic

<https://www.hackerearth.com/es/challenge/competitive/september-circuits-17/algorithm/number-of-divisors-3-4f073391/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static int MOD = 1000000007;

static int MAX = 1000000 + 5;

static long[] A = new long[MAX];

static long mpow(long a, long b, long c)

{

long res = 1;

while (b > 0)

{

if (b % 2 != 0) res = (res \* a) % c;

b >>= 1;

a = (a \* a) % c;

}

return res;

}

static void Main(string[] args)

{

//long x = int.Parse(Console.ReadLine());

//long[] A = Array.ConvertAll(Console.ReadLine().Trim().Split(' '), e => long.Parse(e));

long pro = 1, x, ans = 1, pro1;

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

//for (int i = 0; i < n; i++)

//{

// A[i] = long.Parse(Console.ReadLine());

//}

long[] A = Array.ConvertAll(Console.ReadLine().Trim().Split(' '), e => long.Parse(e));

//REP(i, 0, n, 1) {

// cin >> A[i];

// pro = (pro \* (A[i] + 1)) % MOD;

// }

for (int i = 0; i < n; i++)

{

pro = (pro \* (A[i] + 1)) % MOD;

}

for (int i = 0; i < n; i++)

{

pro1 = (pro \* mpow(A[i] + 1, MOD - 2, MOD)) % MOD;

x = (A[i] \* (A[i] + 1)) % MOD;

x = (x \* mpow(2, MOD - 2, MOD)) % MOD;

ans = (ans \* ((pro1 \* x + 1) % MOD)) % MOD;

}

Console.WriteLine(ans);

Console.ReadLine();

}

}

}