**Archery**

Attempted by: **602**

/

Accuracy: **66%**

/

Maximum Score: **30**

/

2 Votes

Tag(s):

Easy-Medium, Math

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

NN archers are shooting arrows at targets. There are infinite targets numbered starting with 11. The ithith archer shoots at all targets that are multiples of kiki.

Find the smallest target that is hit by all the archers.

**Input**

The first line contains an integer TT - the total no. of testc ases.

TT test cases follow. Each test case is of the following format:

The first line contains a natural number - NN - the number of archers. The second line contains NN space-separated integers, where the ithith integer denotes the value of kiki for the ithith archer.

**Output**

For each test case, print the **smallest target that is hit by all archers** on a new line.

**Constraints**

1≤T≤51≤T≤5

1≤N≤151≤N≤15

1≤ki≤481≤ki≤48

**SAMPLE INPUT**

1

3

2 3 4

**SAMPLE OUTPUT**

12

**Explanation**

The first archer shoots at targets 2, 4, 6, 8, 10, 12, 14, ...

The second archer shoots at targets 3, 6, 9, 12, ...

The third archer shoots at targets 4, 8, 12, 16, 20, ...

The smallest target at which all archers shoot is 1212.

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala 2.11.8, Swift, Visual Basic

<https://www.hackerearth.com/practice/math/number-theory/basic-number-theory-1/practice-problems/algorithm/archery-1/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static long gcd(long a, long b)

{

if (b == 0)

return a;

return gcd(b, a % b);

}

// Returns LCM of array elements

static long findlcm(long[] arr, int n)

{

// Initialize result

long ans = arr[0];

// ans contains LCM of arr[0],..arr[i]

// after i'th iteration,

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

ans = (((arr[i] \* ans)) /

(gcd(arr[i], ans)));

return ans;

}

static void Main(string[] args)

{

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

while (t-- > 0)

{

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

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

Console.WriteLine(findlcm(arr, arr.Length));

}

Console.ReadLine();

}

}

}