**Maximise GCD**

Attempted by: **602**

/

Accuracy: **37%**

/

Maximum Score: **20**

/

4 Votes

Tag(s):

Easy, Math, Number Theory

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

Let us denote by G−ValueG−Value the GCD of all elements of an array. Let us call the prefix of size PP (P≥1P≥1) of an array AA as another array which contains only the first PP elements of the array AA.

You are given an array AA of NN integers. You need to remove some prefix of the array so that the G−ValueG−Value of the remaining array (after removing the prefix) is maximised. You need to print the size of the prefix that should be removed. If there are several possible answers, print the length of minimum size prefix.

**Input:**

First line contains an integer NN, denoting the size of array AA. Next line contains NN space separated integers denoting the array AA.

**Output:**

Print a single integer, the size of the prefix that should be removed so that the G−ValueG−Value of the remaining array is maximised. If there are several possible answers, print the length of minimum size prefix.

**Constraints:**

2≤N≤1032≤N≤103

1≤Ai≤1061≤Ai≤106

**SAMPLE INPUT**

5

4 8 6 9 3

**SAMPLE OUTPUT**

2

**Explanation**

If we remove size 22 prefix, the G−ValueG−Value of remaining array [6, 9, 3] becomes 33.

**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++, 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/problem/algorithm/maximise-gcd-4126af7b/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static long gcd(long a, long b)

{

if (a == 0)

return b;

return gcd(b % a, a);

}

static void Main(string[] args)

{

//long[] arr = { 4, 8, 6, 9, 3 };

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

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

long GCD = 0;

long max\_gcd = 0;

int min\_ind = arr.Length - 1;

for (int i = arr.Length - 1; i >= 0; i--)

{

GCD = gcd(GCD, arr[i]);

if (GCD >= max\_gcd)

{

max\_gcd = GCD;

min\_ind = i;

}

}

Console.WriteLine(min\_ind );

Console.ReadLine();

}

}

}