[Home](https://www.codechef.com/) » [Compete](https://www.codechef.com/contests/) » [Code-e-Mania](https://www.codechef.com/CEM2018?order=desc&sortBy=successful_submissions)  » The simplest problem on earth

**The simplest problem on earth**Problem Code: **IT001**

******

[Tweet](https://twitter.com/share)

**Problem description.**

Given an array of size N .You need to find maximum subarray (not necessarily contiguous) which is having its GCD (greatest common divisor) as K .return its length as output.

**Input**

* The first line of the input contains a single integer T denoting the number of test cases. The description of T test cases follows
* The first line of each test case contains a single integer N.
* The second line contains N space-separated integers A1, A2, ..., AN denoting array content.
* the next line contains K.

**Output**

* For each test case, print a single line containing one integer — the maximum size of subarray with GCD K.
* if no such subarray exists print "0" without Quotes

**Constraints**

1<=T<=10000  
1<=N<=10000  
1<=arr[i]<=50000  
1<=k<=50000

**Example**

**Input:**

2

3

1 2 3

3

5

25 500 11 3 15

5

**Output:**

1

3

Author:[3★yash\_khatri](https://www.codechef.com/users/yash_khatri)

Date Added:23-03-2018

Time Limit:1 secs

Source Limit:50000 Bytes

Languages:C, CPP14, JAVA, PYTH, PYTH 3.5, PYPY, CS2, PAS fpc, PAS gpc, RUBY, PHP, GO, NODEJS, HASK, rust, SCALA, swift, D, PERL, FORT, WSPC, ADA, CAML, ICK, BF, ASM, CLPS, PRLG, ICON, SCM qobi, PIKE, ST, NICE, LUA, BASH, NEM, LISP sbcl, LISP clisp, SCM guile, JS, ERL, TCL, kotlin, PERL6, TEXT, SCM chicken, CLOJ, COB, FS

<https://www.codechef.com/CEM2018/problems/IT001>

**Solution: 17953422**

**CodeChef submission 17953422 (JAVA)** [**plaintext**](https://www.codechef.com/viewplaintext/17953422)[**list**](https://www.codechef.com/status/IT001,apoorvsaxena/)**. Status: AC, problem IT001, contest CEM2018. By apoorvsaxena (apoorvsaxena), 2018-03-24 22:53:30.**

* import java.util.Scanner;
* import java.util.Vector;
* **class** IT001 {
* static long findGCD(Vector<Long> v)
* {
* long result = v.get(0);
* for (int i=1; i<v.size(); i++)
* result = gcd(v.get(i), result);
* **return** result;
* }
* static long gcd(long a, long b)
* {
* if (a == 0)
* **return** b;
* **return** gcd(b%a, a);
* }
* **public** static void main([**String**](http://www.google.com/search?q=allinurl%3AString+java.sun.com&bntl=1) args[]){
* int T;
* Scanner scanner = **new** Scanner([**System**](http://www.google.com/search?q=allinurl%3ASystem+java.sun.com&bntl=1).in);
* T = scanner.nextInt();
* for (int i=0; i<T; i++){
* int n, k;
* n = scanner.nextInt();
* Vector<Long> v = **new** Vector<>();
* Vector<Long> d = **new** Vector<>();
* long counter = 0;
* for (int j=0; j<n; j++){
* long temp = scanner.nextLong();
* v.add(temp);
* }
* k = scanner.nextInt();
* for (int j=0; j<n; j++){
* if (v.get(j)%k == 0){
* d.add(v.get(j));
* }
* }
* if(findGCD(d) == k)
* [**System**](http://www.google.com/search?q=allinurl%3ASystem+java.sun.com&bntl=1).out.println(d.size());
* else
* [**System**](http://www.google.com/search?q=allinurl%3ASystem+java.sun.com&bntl=1).out.println(0);
* }
* }
* }