Bricks and Building

Max. Marks: 100

An old planet called Alpha has been found but it is destructed completely. Archaeologists after expedition found out a map that depicts all the buildings that were in Alpha. In the map following facts were stated:

- Buildings were made of the identical bricks and all the bricks were equal sized.
- All the buildings and bricks were 2D rectangles of some height and equal width.
- A building is formed by putting bricks one on the another. The height of the building is the sum of heights of bricks. None of the bricks used are ever broken or rotated.

Now you have q queries.

Each query contains an integer k. For every query you have to print the count of buildings that can be made if you had infinite number of bricks of size k adhering to the above rules.

Constraints

$1 \leq N \leq 10^5$
$1 \leq A[i] \leq 10^5$
$1 \leq q \leq 10^5$
$1 \leq K \leq 10^5$

Input

First line contains a number N as input denoting the total number of buildings in the planet Alpha. Next N lines contain one integer each denoting the height of each of the buildings. Next line contains a number q as input that denotes the total queries to be asked. For each of the q queries next q lines contain an integer k each.

Output

Print the answer as per the description above for each query in a new line.

SAMPLE INPUT

```
4
5
8
10
8
```

```
1
2
```

SAMPLE OUTPUT

```
3
```

Explanation

In the sample you can see that out of the given four buildings we can make the last 3 buildings using bricks of height 2.

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

CODE EDITOR

Enter your code or Upload your code as file. Save C (gcc 5.4.0)

```c
#include <stdio.h>

int main()
{
    int n,Que,q;
    scanf("%d",&n);
    int a[n];
    for(int i=0;i<n;i++)
        scanf("%d",&a[i]);
    scanf("%d",&Que);
    while(Que--){
        int sum=0;
        scanf("%d",&q);
        for(int i=0;i<n;i++)
            if(a[i]%q!=0)
                sum++;
        printf("%d\n",n-sum);
    }
    return 0;
}

```

1:1

Provide custom input

Press Ctrl-space for autocomplete suggestions.

COMPILE & TEST SUBMIT

Tip: You can submit any number of times you want. Your best submission is considered for computing total score.

Your Rating: