

AVAILABLE
LESSONS:*Lesson 1*
Iterations*Lesson 2*
Arrays*Lesson 3*
Time Complexity*Lesson 4*
Counting
Elements*Lesson 5*
Prefix Sums*Lesson 6*
Sorting*Lesson 7*
Stacks and
Queues*Lesson 8*
Leader*Lesson 9*
Maximum slice
problem*Lesson 10*
Prime and
composite
numbers*Lesson 11*

RESPECTABLE

CountNonDivisible

START

Calculate the number of elements of an array that are not divisors of each element.

Programming language: C++ ▼

You are given a non-empty zero-indexed array A consisting of N integers.

For each number $A[i]$ such that $0 \leq i < N$, we want to count the number of elements of the array that are not the divisors of $A[i]$. We say that these elements are non-divisors.

For example, consider integer $N = 5$ and array A such that:

$A[0] = 3$
 $A[1] = 1$
 $A[2] = 2$
 $A[3] = 3$
 $A[4] = 6$

For the following elements:

- $A[0] = 3$, the non-divisors are: 2, 6,
- $A[1] = 1$, the non-divisors are: 3, 2, 3, 6,
- $A[2] = 2$, the non-divisors are: 3, 3, 6,
- $A[3] = 3$, the non-divisors are: 2, 6,
- $A[4] = 6$, there aren't any non-divisors.

Write a function:

```
vector<int> solution(vector<int> &A);
```

that, given a non-empty zero-indexed array A consisting of N integers, returns a sequence of integers representing the amount of non-divisors.

The sequence should be returned as:

- a structure Results (in C), or
- a vector of integers (in C++), or
- a record Results (in Pascal), or
- an array of integers (in any other programming

**Sieve of
Eratosthenes***Lesson 12*Euclidean
algorithm*Lesson 13*Fibonacci
numbers*Lesson 14*Binary search
algorithm*Lesson 15*Caterpillar
method*Lesson 16*Greedy
algorithms*Lesson 17*Dynamic
programming*Lesson 90*Tasks from
Indeed Prime
2015 challenge*Lesson 91*Tasks from
Indeed Prime
2016 challenge*Lesson 92*Tasks from
Indeed Prime
2016 College
Coders
challenge*Lesson 99*

language).

For example, given:

 $A[0] = 3$ $A[1] = 1$ $A[2] = 2$ $A[3] = 3$ $A[4] = 6$

the function should return [2, 4, 3, 2, 0], as explained above.

Assume that:

- N is an integer within the range [1..50,000];
- each element of array A is an integer within the range [1..2 * N].

Complexity:

- expected worst-case time complexity is $O(N \cdot \log(N))$;
- expected worst-case space complexity is $O(N)$, beyond input storage (not counting the storage required for input arguments).

Elements of input arrays can be modified.

Copyright 2009–2016 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

**For
programmers**

Lessons Challenges
Terms FAQ

For companies

Tour About
Pricing us
Blog Jobs
Privacy Terms
Cookies
API

**Sign up for our
newsletter:**

Information about
upcoming challenges,
solutions and lessons
directly in your inbox.

Social:

[f](#) [t](#) [in](#)

Contact us:

For customer support
queries:

UK +44 (0) 208 970
78 68

US 1-415-466-8085
support@codility.com

For sales queries:

UK +44 (0) 208 970
78 67

US 1-415-466-8085
sales@codility.com

© 2009–2016 Codility Ltd., registered in England and Wales (No. 7048726). VAT ID
GB981191408. Registered office: 107 Cheapside, London EC2V 6DN