

UPCOMING
CHALLENGES:**indeed[®]prime**
Indeed Prime
Challenge**1** days **6** hours
leftCURRENT
CHALLENGES:

Scandium 2016

PAST CHALLENGES

indeed[®]prime
Indeed Prime 2**indeed[®]prime**
Indeed Prime

Calcium 2015

Kalium 2015

Argon 2015

Chlorum 2014

Sulphur 2014

Phosphorus
2014

Silicium 2014

RESPECTABLE

NumberOfDiscIntersections

START

Compute the number of intersections in a sequence of discs.

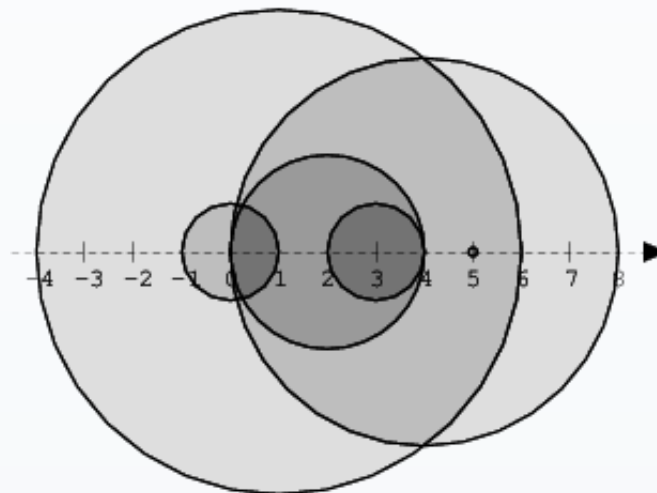
Programming language:

We draw N discs on a plane. The discs are numbered from 0 to $N - 1$. A zero-indexed array A of N non-negative integers, specifying the radii of the discs, is given. The J -th disc is drawn with its center at $(J, 0)$ and radius $A[J]$.

We say that the J -th disc and K -th disc intersect if $J \neq K$ and the J -th and K -th discs have at least one common point (assuming that the discs contain their borders).

The figure below shows discs drawn for $N = 6$ and A as follows:

$A[0] = 1$
 $A[1] = 5$
 $A[2] = 2$
 $A[3] = 1$
 $A[4] = 4$
 $A[5] = 0$



There are eleven (unordered) pairs of discs that intersect, namely:

- discs 1 and 4 intersect, and both intersect with all the other discs;

Aluminium 2014

Magnesium
2014

Natrium 2014

Neon 2014

Fluorum 2014

Oxygenium 2014

Nitrogenium
2013

Carbo 2013

Boron 2013

Beryllium 2013

Lithium 2013

Helium 2013

Hydrogenium
2013

Omega 2013

Psi 2012

Chi 2012

Phi 2012

Upsilon 2012

- disc 2 also intersects with discs 0 and 3.

Write a function:

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

that, given an array A describing N discs as explained above, returns the number of (unordered) pairs of intersecting discs. The function should return -1 if the number of intersecting pairs exceeds 10,000,000.

Given array A shown above, the function should return 11, as explained above.

Assume that:

- N is an integer within the range [0..100,000];
- each element of array A is an integer within the range [0..2,147,483,647].

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.

Tau 2012
Sigma 2012
Rho 2012
Pi 2012
Omicron 2012
Xi 2012
Nu 2011
Mu 2011
Lambda 2011
Kappa 2011
Iota 2011
Theta 2011
Eta 2011
Zeta 2011
Epsilon 2011
Delta 2011
Gamma 2011
Beta 2010
Alpha 2010

For programmers

[Lessons](#) [Challenges](#)
[Terms](#) [FAQ](#)

For companies

[Tour](#) [About us](#)
[Pricing](#) [Jobs](#)
[Blog](#) [Terms](#)
[Privacy](#) [Cookies](#)
[API](#)

Sign up for our newsletter:

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

[Sign up](#)

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