cødility

Training center

Check out Codility training tasks

Demo ticket

Session

ID: demoKKXUQB-FYF Time limit: 120 min.

Status: closed

Created on: 2014-03-25 01:34 UTC Started on: 2014-03-25 01:34 UTC Finished on: 2014-03-25 01:35 UTC

Tasks in test

NumberOfDiscIntersections

Task score

100%

03.

05

06 07

08

10.

11. 12. 13.

14. 15.

16. 17.

18. 19.

20.

discs.sort()

return count

low = 0

def find_overlap(discs, x):

high = len(discs) - 1

for i, d in enumerate(discs):

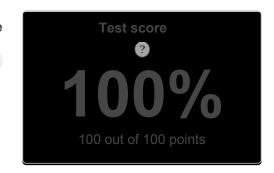
return -1

https://white low <= high:
 mid = (low + high) / 2
 if discs[mid][0] <= x:
 low = mid + 1</pre>

count += find_overlap(discs, d[1]) - i
if count > 10000000:

high = mid - 1
return mid if discs[mid][0] <= x else mid-1

count = 0



1. NumberOfDiscIntersections

Compute intersections between sequence of discs.





Task description

Given an array A of N integers, we draw N discs in a 2D plane such that the I-th disc is centered on (0,I) and has a radius of A[I]. We say that the J-th disc and K-th disc intersect if J ≠ K and J-th and K-th discs have at least one common point.

Write a function:

def solution(A)

that, given an array A describing N discs as explained above, returns the number of pairs of intersecting discs. For example, given N=6 and:

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

intersecting discs appear in eleven pairs of elements:

- 0 and 1
- 0 and 2,
- 0 and 4.
- 1 and 2,
- 1 and 3,
- 1 and 4, • 1 and 5,
- 2 and 3,
- 2 and 4,
- 3 and 4.
- 4 and 5.

so the function should return 11.

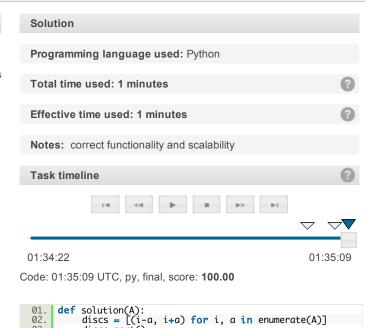
The function should return -1 if the number of intersecting pairs exceeds 10.000.000.

Assume that:

- N is an integer within the range [0..100,000];
- · each element of array A is an integer within the range [0..2147483647].

Complexity:

- expected worst-case time complexity is O(N*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 2009–2014 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

Analysis



O(N * log(N)) or O(N)

test	time	result
example1 example test	0.050 s.	ок
simple1	0.050 s.	OK
simple2	0.050 s.	OK
simple3	0.050 s.	OK
extreme_small empty and [10]	0.050 s.	ок
small1	0.050 s.	OK
small2	0.050 s.	OK
small3	0.050 s.	OK
overflow arithmetic overflow tests	0.050 s.	ок
medium1	0.050 s.	OK
medium2	0.060 s.	OK
medium3	0.090 s.	OK
medium4	0.130 s.	OK
10M_intersections 10.000.000 intersections	0.120 s.	ок
big1	0.050 s.	OK
big2	0.050 s.	ок
big3 [0]*50000	0.050 s.	ок

Training center

© 2009–2014 Codility Ltd., registered in England and Wales (No. 7048726). VAT ID GB981191408. Registered office: 107 Cheapside, London EC2V 6DN