ALL •

segment should be equal to the sum of the capacities of all the interior servers. = sum[1+1, r-1]. In other words, the capacity of the servers at the endpoints of the contiguous subsegment [l, r] of servers is said to be stable if capacity[l] = capacity[r] In an organization, there are n servers, each with a capacity of capacity[i]. A

Find the number of stable sub-segments of length 3 or more.

0

For example, n=5 and capacity=[9, 3, 3, 3, 9].

[9]	1		_			
[9, 3, 3, 3, 9]	[3, 3, 3, 9]	[9, 3, 3, 3]	[3, 3, 9]	[3, 3, 3]	[9, 3, 3]	Segment
9	3	9	w	ω	9	-
9	9	0	ω	ω	6	First & Last Capacity Interior Capacity Sum
Yes	No	No	ONE	Yes	No	Balanced

W

There are 2 stable subsegments: [3, 3, 3] and [9, 3, 3, 3, 9]. Return 2.

Function Description

Complete the function countStableSegments in the editor below.

Language C++20 ::::::: 20 21 > int main() ... be Low. * Complete the 'co * The function a parameter. int countStable The function i 0

Language C++20

0

Constraints

· 1 ≤ n ≤ 3 * 105

2

1 ≤ capacity[i] ≤ 10⁹

Input Format For Custom Testing

::::::::

21 > int main() ...

17 18 19 20

int countStableSeg

14

parameter.

* The function is ex

* The function accep

be LOW .

* Complete the 'counts

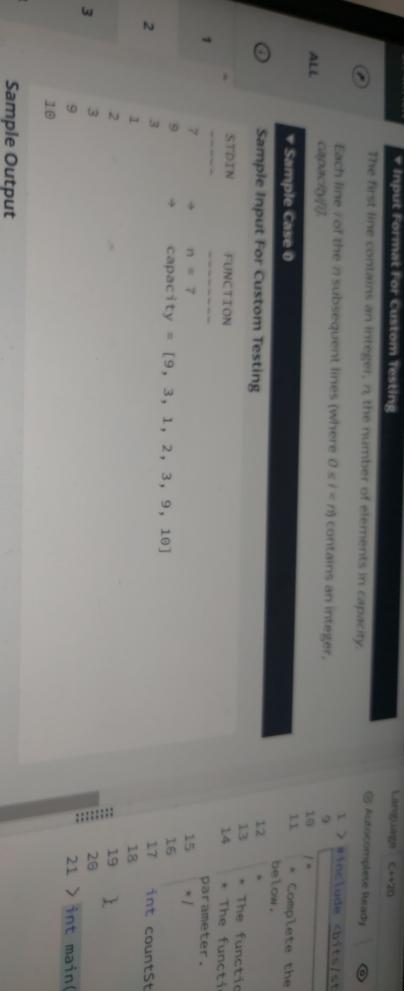
W

▼ Sample Case 0

Sample Input For Custom Testing

Test Resul

IO



int countst

Explanation

The stable segments are [9, 3, 1, 2, 3, 9] and [3, 1, 2, 3].

▼ Sample Case 1

Sample Input For Custom Testing

STDIN 1111111 FUNCTION

capacity = [6, 1, 2, 3, 6]

Test F

0

Sample Output

Explanation

The stable segments are [9, 3, 1, 2, 3, 9] and [3, 1, 2, 3].

▼ Sample Case 1

2

Sample Input For Custom Testing

::::::::

19

21 > int main

18

int count

below.

* The func * The func parameter.

ω

capacity = [6, 1, 2, 3, 6]

4

Sample Output

H

Explanation

The entire array is a stable segment.

Test Res

9