Equilibrium Index of an array

01 December 2023 02:14 PM

You are given an array **A** of integers of size **N**.
Your task is to find the equilibrium index of the given array
The equilibrium index of an array is an index such that the sum of elements at lower indexes is equal to the sum of elements at higher indexes.
If there are no elements that are at lower indexes or at higher indexes, then the corresponding sum of elements is considered as 0.
Note:

- Array indexing starts from 0.
- If there is no equilibrium index then return -1.
- If there are more than one equilibrium indexes then return the minimum index.

From <https://www.scaler.com/academy/mentee-dashboard/class/76354/assignment/problems/12826>

Problem Constraints
1 <= N <= 105
-105 <= A[i] <= 105

Input Format
First arugment is an array A .

Output Format
Return the equilibrium index of the given array. If no such index is found then return -1.

Example Input
Input 1:
A = [-7, 1, 5, 2, -4, 3, 0]

Input 2:
A = [1, 2, 3]

Example Output
Output 1:
3

Output 2:
-1

[-7, 1, 5, ...
0 1 2 3

i	Sum
0	
1	-
2	
3	-
4	1
5	-3
6	0

[1, 2, 3]
0 1 2

i	Sum of
0	0
1	1
2	3

Bruteforce : iterate over each Indx & for each Indx Calucate Sum of it lower Indx elements & higher Indx elements & check it equality.

T.C. $O(n^2)$
S.C $O(1)$

Soln 2 : Using Prefix Sum

[-7, 1, 5, 2, -4, 3, 0] => [-7, -6, -1, 1, -3, 0, 0]
0 1 2 3 4 5 6 — Prefix Sum array

i	Sum lower Indx	Sum high Indx
0	0	7

1	-7	6 (0 - (-6))
2	-6	1 (0 - (-1))
3	-1	-1 (0 - 1)
4	1	3 (0 - (-3))
5	-3	0 (0 - 0)
6	0	0 (0 - 0)

sum of

=

Sum of he

= PS[

$$[\underset{0}{1}, \underset{1}{2}, \underset{2}{3}] = [1, 3, 6] = PS$$
$$\quad\ 0\ \ 1\ \ 2$$

i	Sum lower Indx	Sum higher Indx	
0	0	5	(6 - 1)
1	1	3	(6 - 3)
2	3	0	(6 - 6)

— x —— x — x —— x — x —

Pseudocode :-

```
A = [ ]  // Input
n = A.size();
PS = [n-1];
// Create PS array
PS[0] = A[0];
for (int i=1 ; i<n; i++) {
    PS[i] = PS[i-1] + A[i];
```

```
    }

    // iterate ps array to check low sum index & high
    for (int i=0; i<n; i++) {
        loSum = (i==0) ? 0 : ps[i-1];
        hiSum = ps[n]-ps[i]
        if (loSum == hiSum) {
            return i;
        }
    }

    return -1;
}
```

T.C. = O(n)

S.C = O(n) : for prefix sum array.

Optimization :

If we can update the input array to prefix s

T.C = O(n)

S.C = O(1)

Edge case : If array is empty return -1;