ELogo C C C C C C C C C C C C C C C C C C	
STUDENT REPORT	,
STOPENT REPORT	AOO.
STUDENT REPORT Name Of the Ball Student Age 12 Chook 3 Ba	2300
DETAILS 1364 1300 3842 100 3842 100 3842	0
Name Ook State Cook St	28R
AKANKSHA N	300
Roll Number 38823CD004	20
3BR23CD004 EXPERIMENT Title EQUILIBRIUM Description Description	8823C
EXPERIMENT A Title EQUICIBRIUM Description A 36 P2 3 CD O A 36	× ×
Title 3CDOW 38F273CDOW 38F272CDOW	~ A
EQUILIBRIUM (DOUNDER) ARTICOLONIA ARTICOLO	3000
Clock Strain Strain Cook Strain Cook Strain Cook Strain Cook	L.
Description 38th 31 Cook 38th 3 Cook 38th	88
You are given an array A of N integers. An equilibrium position is a position where the sum of all integers on its left is equal to the	200A 3BP
sum of all integers on its right in the array A. Print the index of the equilibrium position.	
sum of all integers on its right in the array A. Print the index of the equilibrium position. Note:For any given array there is only a single equilibrium position, if no equilibrium position is found then print "NOT FOUND" without quotes.	38R23C
The array is 1 indexed	3BR
Input Format:	230004
	23
The imput consists of two lines.	
The second line contains N space-separated integers denoting the elements of the array A.	DOOA 388
	500
	<
Print the index of the equilibrium position. If no index is found, print "NOT FOUND"	×38R23C
That the index of the equilibrium position. If no index is found, print NOT FOOND	*
	É
24733	P. B. L. B. B.
Sample Output	900
Sample Output	269
	Osphi,
Source Code: 3CDO 3BR23 CDO A3V CDO A3V CDO A3V CDO A3V CDO A3V M3BRAR ABARTAL BA	A.
382 (100h) 232CO (23EEH) 30 (23CO) (23EEH) 30 (23	30
Source Code: 3CDO 43BR23CDO 43FR23CDO 43FR23CD	42386°
38 Paris Sook 3	5
Source Code: 3CDOA 3BR23CDOA 3BR23CD	(Digg
Source Code: 30 Part of the American Am	E BODO

```
def find_equilibrium_position(N, A):
       total_sum = sum(A)
       left_sum = 0
       for i in range(N):
           right_sum = total_sum - left_sum - A[i]
           if left_sum == right_sum:
               return i + 1
           left_sum += A[i]
       return "NOT FOUND"
   # Input reading
   N = int(input())
   A = list(map(int, input().split()))
                                                                                                         38R13C100A38R135
   result = find_equilibrium_position(N, A)
   print(result)
RESULT
 5 / 5 Test Cases Passed | 100 %
```