,n,*	□ Logo	Ds.
ECOAA 3BED 35	COLA 35 COLA 36 COLA 3	1
COM 35	A STATE OF THE STA	- SECON,
DETAILS Name G SANDHY.	ASHRISECOAA 3BRISECOAA	3R23
DETAILS	A SEL COMM SELSE MASSEL LECOMM SELSE MASSEL LECOMM	~
Name	38200 3ECON 3BRD3T	CAA 3b
	TA NEDD I	ſ
Roll Number	1,3°	~ .
3BR23EC04	144	38273
EXPERIMEN	ASBER SECONN SERVICONN SER	A
Title	Secon 38Ergs 34Ergs 3Ergs	Q.A.
EQUILIBRIUM	M COAA 223EC A3BELL COAA	273400
3 ^E COM	OFFISE OF	ρ,
Description	A STATE OF THE STA	38
You are give	ven 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 sum	COAA 38
of all intege	ers on its right in the array A. Print the index of the equilibrium position.	
Note:For an quotes.	any given array there is only a single equilibrium position, if no equilibrium position is found then print "NOT FOUND" without	, A 3BR23K
The array i	is 1 indexed.	136°
No. of the second secon		
Input Form	nat:	a ECOA
	consists of two lines:	1823
25		á
The secon	nd line contains N space-separated integers denoting the elements of the array A.	ECOAA 3B
	be read from the STDIN by the candidate	
Output For		~ ~
Print the in	ndex of the equilibrium position. If no index is found, print "NOT FOUND"	, A 3BR234
		, Lx
Sample Inp		- OF
24733		FEBINA.
	utput	3~
Sample Ou		200
15.		SKY OSE,
Source Code	et store sto	,
30	SECONT SERVICE CONTRACTOR SECONTRACTOR SECON	a Received
	e: 3ECU AA 3ECOAA 3ECOA	1385 J. 2
	25/3/2 1/2 3/2 1/2 2/2 1/2 2/2 2/2 1/2 2/2 2/2 2/2 2	js.
	3° Series	28/A
	et. 34C ^{CC} 3HR ^{23ECO} AA 3HR ²³	2350

```
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()))
   result = find_equilibrium_position(N, A)
   print(result)
RESULT
 5 / 5 Test Cases Passed | 100 %
```