	DLogo C X♥ C X	,Q~
Jech	DETAILS Name Replace Tree Tree Tree Tree Tree Tree Tree Tr	OBJETH BR
3,	Techiest Cold Stechies Stool Material Stool Callette Stool Callett	62 44
<u>_</u>	DETAILS SEED OF THINK SEEDS SE	chi esto
062	DETAILS Name, Notice to the feeting of the property of the pr	oo2 Tehre
lech.co		TEM!
	TEMPBTech-CSE062	TEMPO
E	EXPERIMENTS OF THE PROPERTY OF	cst062
S EW.	Title Atechic CHOOL MARKED CEPTOR THANKS CHOOL COLLEGE	1002
	EQUILIBRIUM COS AFEET SEOON METER COSTON COS	, ech. CSt
en csto	TEMPBTech-CSE062 EXPERIMENT Title EQUILIBRIUM Description 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 a sum of	EMPBIL
30	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 sum NRBT ech
EMPBI	Note :For any given array there is only a single equilibrium position, if no equilibrium position is found then print "NOT FOUND" (quotes.	without
	The array is 1 indexed.	£ 0
55£067		Č.
J	Input Format:	8Techice
c.X	The input consists of two lines:	
RBTecx	The first line contains an integer denoting N.	62 TEMPS
	The decond line contains it opace departies integers denoting the elements of the unity / i.	,62
5062	Input will be read from the STDIN by the candidate Output Format:	(
500	Print the index of the equilibrium position. If no index is found, print "NOT FOUND"	n.cski)
		2 ^C
Tech.	5	
	24733	FIFWHPIT
. 19	Sample Output	ν.
1 Elec	3	Sales of the Sales
	Source Code: Thin Biech, Stool Thin Biech, Stoo	and the contract of the contra
	Source Code: pt & control of the pt	HE MAPINE
	TEMPETERICS TO THE TERMS TO SELOCATE THE PETER SELO	LEW BETT
	Medical Carlos Links Selical Markething Selical Carlos	CHO DA
	The state of the s	SEST COMMENT
	TELLY, TECHTO CONTENTS OF THE FEET OF THE	C. C. S. C. S. C.

```
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 %
```