	Logo	
, (£)	ETAILS  Name 26 16 16 16 16 16 16 16 16 16 16 16 16 16	
STUDENT REPORT  PASHWANTH I		
	Testil 1030 The Refer THO30 HALE THE STATE OF THE STATE O	) \{\langle_{\chi}\}
(¢, <b>D</b> )	ETAILS SEET SEETS SEETS SEETS SEEDS	0
530	ETAILS  Name Report Helion Hel	LEE03
	YASHWANTH J	,5'
(echitet)	Roll Number, NY SHEET SOUTH SEET SOUTH SEED	cò
180.	TEMPBTech-EEE030	MPBTO
E)	TEMPBTech-EEE030  XPERIMENT, 230 Hand to collect the collect the collect to a societion where the sum of all interess as its left to sound to the collect to sound to the coll	<i>&gt;</i>
EMPTit	tle  EQUILIBRIUM  Description Note of the	~ <<
2	EQUILIBRIUM NELLOSO TENTE CENTELLOSO SOLITANDO SE ESTELLES SOLITANDO SOLITAN	EEO33
035	oft setes the so the steel set	
3n.EEELO3S	Description At Service	rec'n.
	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	RATES
EMPBTEC	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	
EWE	quotes.	1030 TEM
	The array is 1 indexed.	0,0
EFFO30		¿\$
	Input Format:	Kechitlet
<u> </u>	The input consists of two lines:	,
RATECHI	The first line contains an integer denoting N.	, PE
14.	The second line contains N space-separated integers denoting the elements of the array A.	OTEMPÉ
4.3	input will be read from the STDIN by the candidate	
£030 TEN	Output Format:	£ 60°
	Trine the index of the equilibrium position. If no index is round, print 1901 Footb	CHEEFO?
	Sample Input	
STECKLER	5	NB C
	24733	Eleky.
LEWLB	Sample Output	4
^*	3	(4. E.
	Source Code: Accident	
	Source Code: Pate Country of the Cou	%
	The state of the s	White Service
	The fee, fifting the fee, fitting the fitt	v.
	Source Code: Hand to child the child the child to child the child to child the child to child the child	
	TENTO ELIPTIC COLLEGE	FORM ,
		/

```
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 %
```