ELogo ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~	
STUDENT REPORT	3
Ecoso STUDENT REPORT STUDENT REPORT Name C NAAZNEEN	
Name cost age of the cost of t	8
C NAAZNEEN	
Roll Number 1984 1985 1985 1985 1985 1985 1985 1985 1985	4
3BR23EC030	5
EXPERIMENT, SALOSOS SALOS SALOSOS SALOS SALOS SALOS SALOS SALOSOS SALOS SALO	
EQUILIBRIUM (CO) ARPANICO ARPA	3
EQUILIBRIUM ACO ARANI AR	
3BR23EC030 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 the sum	\$
Tou are given an array A or in integers. An equilibrium position is a position where the sum or an integers on its left is equal to the sum	
of all integers on its right in the array A. Print the index of the equilibrium position.	
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. The array is 1 indexed.	3
The array is 1 indexed.	
Input Format: The input consists of two lines:	3
The input consists of two lines:	
The input consists of two lines: The first line contains an integer denoting N. The second line contains N space-separated integers denoting the elements of the array A.	Š
The first line contains an integer denoting N. The second line contains N space-separated integers denoting the elements of the array A.	
input will be read from the STDIN by the candidate	
Input will be read from the STDIN by the candidate Output Format: Print the index of the equilibrium position. If no index is found, print "NOT FOUND"	3
Output Format: Print the index of the equilibrium position. If no index is found, print "NOT FOUND"	
Sample Input	
Sample Input 5	35
24733	
Sample Output	
3	200
Source Code: AREA CONTROLL OF THE PROPERTY OF	180 CAS
April 1000 Market	

```
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())
                                                                                                  ~34x13x103034x11
   A = list(map(int, input().split()))
   result = find_equilibrium_position(N, A)
   print(result)
RESULT
 5 / 5 Test Cases Passed | 100 \%
```