	$\mathcal{L}_{\mathcal{S}}^{L}$	75°.
ECO5A3BP	TAILS HARSHINI K	
£C05	38 CO2 38	
D∉	HARSHINI K	3821
3BR273	Jacob Strain Str	38
) N	HARSHINI K	7,005
	2all Number of A A A A A A A A A A A A A A A A A A	
5R13ECO R	3BR23EC054	2823
	DEDIMENT ASERTS COSASS ASERTS	- 30
F. Title	PERIMENT, NO SERVICE S	<
ECO.	NUMBER OF COMBINATIONS LEADING TO A PRODUCT	-13ECO
£	CONDENSITY THOUGHT TO ATTROOP TO	3BR.
N D	3BR23EC054 PERIMENT PERI	3
o ^{lx}	Problem Statement:	¿cosh 38
3R23ECOF	You are given an array arr and a product m. Your task is to find the number of possible unique triplets whose product of	
BRIL	Input Format:	A 3BR23
ECOPA 3R	The second line contains the integer, n The second line contains characteristic distances of the array arr	
	The input will be read from the STDIN by the candidate	,R23Ec05
RISS	Output Format:	
5 A 3BR 23	The output consists of a single integer, i.e. the count of unique triplets having product m.	¿COSA 35
	The output will be matched to the candidate's output printed on the STDOUT	
3R23ECOF	Example:	0
8P		3A 3BR2?
38	7)
,EC05A38	5 3 20 10 1 4 2	(6)
) *	60 Output:	A.S. E.
3BR23	3	-
'S*	Explanation:	C 1/2/2/2
	Product m:60	
	Possible triplets for product m: (5,4,3),(20,3,1), (10,3,2)	-22 ²
	The count of unique triplets is 3.	3/8/3/
S	Source Code: 34PA SECOS AST SOURCE CODE SO	ART REST

```
def count_triplets(arr, n, m):
       unique_triplets = set()
       for i in range(n):
           for j in range(i + 1, n):
               for k in range(j + 1, n):
                   if arr[i] * arr[j] * arr[k] == m:
                       triplet = tuple(sorted([arr[i], arr[j], arr[k]]))
                       unique_triplets.add(triplet)
       return len(unique_triplets)
   # Input Reading
   n = int(input())
   arr = list(map(int, input().split()))
   m = int(input())
   result = count_triplets(arr, n, m)
                                                                                                      ABADALCOSA ABA
   print(result)
RESULT
 6 / 6 Test Cases Passed | 100 %
```