



STUDENT REPORT

DETAILS

Name

D M ISAQ

Roll Number

KUB23CSE033

EXPERIMENT

Title

SUM OF NUMBERS AT PRIME FACTORS

Description

Prime factors of a positive integer are the prime numbers that divide that integer exactly.

Given an array arr of n integers and a positive integer num.

Let's suppose prime factorization of num is: $p^a \times q^b \times r^c \times \dots \times z^f$, where p,q,r...z are prime numbers.

Sum of numbers in array arr at indices of prime factors of number num is: $a \times arr[p] + b \times arr[q] + c \times arr[r] + \dots + f \times arr[z]$.

You are given an array arr of size n and a positive integer num. You are required to calculate the sum of numbers in arr as mentioned above, and print the same.

Note:

- If arr is empty, print -1.
- If prime factor of num not found as indices, print 0.

Input Format:

- The input consists of three lines:
- The first line contains an integer, i.e. n.
 - The second line contains an array arr of length of n.
 - The third line contains an integer num

The input will be read from the STDIN by the candidates.

Output Format:

Print the sum that was mentioned in the problem statement.

Example:

Input:

6
11 21 32 45 1 23
6

Output:

77

Explanation:

$$6=2^1 \times 3^1$$

$$\text{sum}=1*\text{arr}[2]+1*\text{arr}[3]=1*32+1*45=77$$

Source Code:

```
def prime_factors(n):
    factors = {}
    # Check for number of 2s that divide n
    while n % 2 == 0:
        if 2 in factors:
            factors[2] += 1
        else:
            factors[2] = 1
        n //= 2

    # n must be odd at this point, thus a skip of 2 (i.e., check odd numbers)
    for i in range(3, int(n**0.5) + 1, 2):
        while n % i == 0:
            if i in factors:
                factors[i] += 1
            else:
                factors[i] = 1
            n //= i

    # This condition is to check if n is a prime number greater than 2
    if n > 2:
        factors[n] = 1

    return factors

def calculate_sum(arr, num):
    if not arr:
        return -1

    # Get prime factors of the number
    factors = prime_factors(num)

    total_sum = 0
    indices_found = False

    for prime, count in factors.items():
        index = prime - 1 # Convert to 0-based index
        if 0 <= index < len(arr): # Ensure index is within the bounds of arr
            total_sum += count * arr[index]
            indices_found = True

    if not indices_found:
        return 0

    return total_sum

# Input handling
import sys
input = sys.stdin.read
data = input().splitlines()

n = int(data[0].strip()) # Size of the array
arr = list(map(int, data[1].strip().split())) # The array
num = int(data[2].strip()) # The number to factorize

# Calculate the required sum
result = calculate_sum(arr, num)
print(result)
```

RESULT

1 / 5 Test Cases Passed | 20 %