Logo

STUDENT REPORT

DETAILS

Name

A MANOJ

Roll Number

KUB23CSE001

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: $p^a \times q^b \times r^c \times .... \times z^f$,where p,q,r...z are prime numbers.

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

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

sum=1*arr[2]+1*arr[3]=1*32+1*45=77

Source Code:

```
def prime_factors(num):
    """ Returns a dictionary of prime factors with their counts. """
    factors = {}
    # Check for number of 2s that divide num
    while num % 2 == 0:
        if 2 in factors:
            factors[2] += 1
        else:
            factors[2] = 1
        num //= 2
    # Check for odd factors from 3 to sqrt(num)
    for i in range(3, int(num**0.5) + 1, 2):
        while num % i == 0:
            if i in factors:
                factors[i] += 1
            else:
                factors[i] = 1
            num //= i
    # This condition is to check if num is a prime number greater than 2
    if num > 2:
        factors[num] = 1
    return factors

def calculate_sum(arr, num):
    if len(arr) == 0:
        return -1  # Return -1 if the array is empty

    # Get prime factors and their powers
    factors = prime_factors(num)

    total_sum = 0
    found_index = False

    for prime, power in factors.items():
        index = prime
        if index < len(arr):  # Check if the index is valid
            total_sum += power * arr[index]
            found_index = True

    if not found_index:
        return 0  # If no valid indices were found

    return total_sum

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

n = int(data[0])
arr = list(map(int, data[1].split())) if n > 0 else []
num = int(data[2])

# Calculating and printing the result
result = calculate_sum(arr, num)
print(result)
```

RESULT

4 / 5 Test Cases Passed | 80 %