Logo

STUDENT REPORT

DETAILS

Name

K DHANUSH KUMAR

Roll Number

KUB23CSE055

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(n):
    """ Return a dictionary of prime factors and their counts. """
    factors = {}
    while n % 2 == 0:
        if 2 in factors:
            factors[2] += 1
        else:
            factors[2] = 1
        n //= 2

    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

    if n > 2:
        factors[n] = 1

    return factors

def main():
    import sys
    input = sys.stdin.read
    data = input().strip().splitlines()

    n = int(data[0])
    if n == 0:
        print(-1)
        return

    arr = list(map(int, data[1].split()))
    num = int(data[2])

    factors = prime_factors(num)

    weighted_sum = 0
    for prime, count in factors.items():
        index = prime
        if 0 <= index < n:
            weighted_sum += count * arr[index]

    print(weighted_sum if weighted_sum > 0 else 0)

if __name__ == "__main__":
    main()
```

RESULT

4 / 5 Test Cases Passed | 80 %