



STUDENT REPORT

DETAILS

Name

RUDRA PRAKASH TIWARI

Roll Number

3BR23ME016

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:

Source Code:

```
import math
def isprime(n):
    if n <= 1:
        return False
    for i in range(2,int(math.sqrt(n)) + 1):
        if n % i == 0:
            return False
    return True
N=int(input())
if N==0:
    print(-1)
    exit()
A=list(map(int,input().strip().split()))[:N]
P=int(input())
numsP = {}
for i in range(2, P + 1):
    while isprime(i) and P%i == 0:
        if i in numsP:
            numsP[i] += 1
        else:
            numsP[i] = 1
    P //=i
answer =0
for key,value in numsP.items():
    if key < N:
        answer += value*A[key]
    else:
        answer = 0
        break
print(answer)
```

```
6=21 x 31

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

RESULT

3 / 5 Test Cases Passed | 60 %