Array Significance

DiPS CodeJam 23

Prompt

Pranav and Prithvi from CodeJam 22 are back, and are now working on a mathematical proof. They have determined the *significance factor function* $s(x[])$ of an array to be as follows:

$$\left\lfloor \frac{\text{number of primes in } x[]}{\text{highest common factor of the composite numbers in } x[]} \cdot \text{length of } x[] \right\rfloor$$

square brackets indicate Greatest Integer Function
When there are no primes in $x[]$ or $x[]$ has no composite elements, $s(x[]) = 0$.
Given a list $n[]$, determine the significance factor of the array.

Input Format

The first and only line of input contains a space-separated list $n[]$

Output Format

Your output must have a single line containing $s(n[])$.

Constraints

$1 \leq$ length of $n[] \leq 10^5$ and $1 \leq n[i] \leq 100$

Sample Input/Output

Input	Output
2 3 4 5 6 7 8 9	32

Sample Program

```
from math import *

n=list(map(int, input().strip().split()))

number_of_primes = 0
composites = []

def isprime(x):
  for i in range(2, x//2+1):
    if x%i == 0:
      return False
  return True

def list_gcd(nums):
```

```
  if len(nums) == 1:
    return nums[0]

  div = gcd(nums[0], nums[1])

  if len(nums) == 2:
    return div

  for i in range(1, len(nums) - 1):
    div = gcd(div, nums[i + 1])
    if div == 1:
      return div

def GIF(x):
  return int(floor(x))

for i in n:
  if isprime(i):
    number_of_primes+=1
  else:
    composites.append(i)

s=0

if number_of_primes!=0 and len(composites)!=0:
  s=(number_of_primes/list_gcd(composites))*len(n)

print(GIF(s))
```