 Some prime numbers can be expressed as a sum of other consecutive prime numbers.  
  
  
  
  
For example  
  
5 = 2 + 3,  
  
17 = 2 + 3 + 5 + 7,  
  
41 = 2 + 3 + 5 + 7 + 11 + 13.  
  
Your task is to find out how many prime numbers which satisfy this property are present in the range 3 to N subject to a constraint that summation should always start with number 2.  
  
Write code to find out the number of prime numbers that satisfy the above-mentioned property in a given range.  
  
  
  
  
Input Format: First line contains a number N  
  
  
  
  
Output Format: Print the total number of all such prime numbers which are less than or equal to N.  
  
  
  
  
Constraints: 2<N<=12,000,000,000

num = int(input())

arr = []

sum = 0

count = 0

if num > 1:

for i in range(2, num + 2):

for j in range(2, i):

if i % j == 0:

break

else:

arr.append(i)

def is\_prime(sum):

for i in range(2, (sum // 2) +2):

if sum % i == 0:

return False

else:

return True

for i in range(0, len(arr)):

sum = sum + arr[i]

if sum <= num:

if is\_prime(sum):

count = count + 1

print(count)

