The following are the o/p of the following code:

\\\\

import time

start\_time = time.time() # Start tracking time

from sympy import divisors; from functools import lru\_cache

cached\_divisors = lru\_cache()(divisors)

def A071324(n): return sum(d if i%2==0 else -d for i, d in enumerate(reversed(cached\_divisors(n))))

def candidates\_analysis(n):

if n % 30 == 24 or n % 12 == 8 or n % 30 == 20:

return True

else:

return False

# Parse only even numbers in this.

def mod\_check(n):

if n % 30 == 24:

return "n = 24 (mod 30) "

elif n % 12 == 8:

return " n = 8 (mod 12) "

elif n % 30 == 20:

return " n = 20 (mod 30)"

# Main search loop.

count = 0

for n in range(6818712836,(10\*\*15)+1,2):

count += 1

if candidates\_analysis(n):

if A071324(n) == A071324(n+1):

print("Result! n = ",n,"A071324(n) = ",A071324(n),"Modular relation -",mod\_check(n))

# Progress report

if count % 10\*\*6 == 0:

elapsed = time.time() - start\_time

print("Checked up to n =", n, "Time elapsed (sec):", round(elapsed, 2))

\\\\\\\

Results - o/p