```python

from math import log


def run(primeList: list) -> list:
  """
  Chebyshev's Theta Function.
  Returns a sorted list containing the log transformed product of the primorial at each prime in the
  given list.
  """

  # list for storing the product at each nth prime
  products = list()

  # initiating var to store current Theta(x)
  lastVal = 0

  # iterating through all primes in the list
  for prime in primeList:

    # getting sum of logs
    # using log laws, we know log(n) + log(m) == log(nm)
    current = log(prime) + lastVal

    # storing to list
    products.append(current)

    # updating product
    lastVal = current

  return products


```

Algorithm for calculating θ(x) for a list of prime numbers using Chebyshev's first function, returning a list of all θ(x) values in ascending order.

Chebyshev's First Function is defined as

$\theta(x) = \sum_{p \leq x} \log(p)$

where p is a prime number.

Because this function is a sum, the provided list of primes must include all primes less than or equal to the greatest prime in the given list. If all primes are not included, the resulting θ(x) will not be accurate.

Because this is an arithmetic sequence, for each p, we take its natural log and add it to the previously determined θ(x).

The end result is a list of values for θ(x).

Interestingly, the resulting curve produced by this function acts as a lower bound to the set of prime numbers which it represents. The curve is also log-linear, which allows for very loose approximation of where a prime number may fall on $\mathbb{Z}$.

The relative difference between $\theta(x)$ and x approaches 0 as x tends to infinity.