

```

1  from math import log
2
3
4
5  def run(primeList: list) -> list:
6      """
7      Chebyshev's Theta Function.
8      Returns a sorted list containing the log transformed product of the primorial at each prime in the
9      given list.
10     """
11
12     # list for storing the product at each nth prime
13     products = list()
14
15     # initiating var to store current Theta(x)
16     lastVal = 0
17
18     # iterating through all primes in the list
19     for prime in primeList:
20
21         # getting sum of logs
22         # using log laws, we know log(n) + log(m) == log(nm)
23         current = log(prime) + lastVal
24
25         # storing to list
26         products.append(current)
27
28         # updating product
29         lastVal = current
30
31     return products
32

```

Algorithm for calculating $\theta(x)$ for a list of prime numbers using Chebyshev's first function, returning a list of all $\theta(x)$ 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 $\theta(x)$ 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 $\theta(x)$.

The end result is a list of values for $\theta(x)$.

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 prime numbers.

Additionally, $x < \theta(x)$ for all prime numbers in our test range $[0, 1 + e^{10}]$.