Programming C++

Project **Goldbach**

Status: 10. Februar 2023, 13:35

Supervisor: Prof.Dr. G. Haase, gundolf.haase@uni-graz.at

Goldbach's conjecture:
Each even number larger than 3 can be written as sum of two primes (Goldbach's conjecture[1]), i.e., that holds also for all even numbers from $[4, n]$ ($n >= 4$).

1. Incorporate the header mayer_primes.h[2] into your code. The modified code originates from Florian Mayer[3] and generates all primes until n.

2. Write a function `single_goldbach(k)` that counts for a natural number k the number of possible decompositions with 2 primes for k and returns that number to your main code (e.g., $k = 694$ has 19 decompositions).

3. Write a function `count_goldbach(n)` that counts the number of possible decompositions for all even numbers in $[4, n]$ and returns these data.
   Determine in your main code that k with the most decompositions ($n = 100.000 \Longrightarrow k = 99.330$).

4. Measure the run time of your function `count_goldbach(n)` for $n = \{10.000 \quad 100.000 \quad 400.000 \quad 1.000.000 \quad 2.000.000 \quad (10.000.000)\}$.
   Use `system_clock`[4] from the `chrono` functions for timing.

* Write a function similar to the one in 3. but returning all decompositions for all even numbers in the given range.

C++ hints: `vector`, `max_element`, `push_back`

[1] `https://en.wikipedia.org/wiki/Goldbach's_conjecture`
[2] `https://imsc.uni-graz.at/haasegu/Lectures/Math2CPP/Examples/goldbach/mayer_primes.h`
[3] `http://code.activestate.com/recipes/576559-fast-prime-generator`
[4] `http://www.cplusplus.com/reference/chrono/system_clock/now/`