

## 1. Programmieren in C

Abgabe bis 7. Mai, 10:15 GMT+2.

### Aufgabe 0:

Teile den Code aus Aufgaben 0 und 3 vom Übungsblatt 0 geeignet auf .c- und .h-Dateien auf, und passe das Makefile entsprechend an!

### Aufgabe 1:

Tu dies auch für den Code aus Aufgabe 1 von Blatt 0, und einen zugehörigen Test!

### Aufgabe 2:

Schreibe eine Funktion `void eratosthenes (size_t n, bool marked[])` für das Sieb des Eratosthenes! `bool` findet sich in `stdbool.h`, `size_t.h` z.B. in `stddef.h`.

Das Sieb des Eratosthenes ist ein Algorithmus zum Finden von Primzahlen. Es werden alle Zahlen ab 2 (bis zu der gewünschten Obergrenze) aufgeschrieben, und dann beginnend mit der kleinsten noch nicht markierten Zahl jeweils für jede solche Zahl alle ihre Vielfachen markiert. Die zuletzt verbleibenden unmarkierten Zahlen sind die Primzahlen. Die Funktion kann dann z.B. so aufgerufen werden:

```
bool marked[8];
eratosthenes(marked, 8);
assert(!marked[2]); // 2 ist prim, und soll somit nicht markiert sein.
assert(marked[6]); // 6 ist nicht prim und soll somit markiert sein.
```

### Aufgabe 3:

Schreibe ein Programm, dass die Funktion aus Aufgabe 2 aufruft, und die gefundenen Primzahlen ausgibt! Nach Start des Programms soll es auf eine Eingabe warten (Zahl gefolgt von Enter), die die größte zu überprüfende Zahl angibt. Die Eingabe kann z.B. per `getchar` eingelesen werden.

Hinweise (gelten für dieses wie auch für alle weiteren Blätter):

- Die Abgabe erfolgt über Subversion in ein Unterverzeichnis Blatt1 (entsprechend bei den folgenden Blättern dann Blatt2, etc).
- Der Code soll modular sein: Umfangreiche oder mehrfach verwendete Teile gehören jeweils in eine eigene Funktion.
- Für jede nichttriviale Funktion soll es einen Test geben. Auch das Verhalten bei eventuellen Grenzfällen (leere Eingabe, etc) soll getestet werden.
- Der Code soll im Google-Stil geschrieben werden (<https://google.github.io/styleguide/cppguide.html>), was im Makefile zu überprüfen ist. Dazu soll das Makefile für das Ziel `checkstyle` für jede Quelldatei `python cpplint.py <Dateiname>` mit `cpplint 1.4.4` aufrufen werden.