Technical University of Munich
School of Engineering and Design
Prof. Dr. Martin Werner

Computational Foundations I (Winter Term 2021/22)
Tutorial 6

Tasks marked with a star like **Optional Task*** are optional. Tasks marked like **Hard Task**+ are given, but it is not expected that you solve them now. It is great if you learn to solve them during the lecture. Go back to them after a few weeks and see your own progress.

Learning Outcome: We are warming up with C/C++ programming environments.

Task 14: C Warmup

Programming in C and C++ is completely different from working with scripting environments like MATLAB. The procedure of code writing, compiling, execution, and debugging is new. Therefore, we encourage you to go through the lecture tutorial tasks related to MATLAB and try to solve some of them in C or C++.

a. **Written Addition:**Given two arrays in code `int x[] = {1,2,3,4}` and `int y[] = {5,6,7,8}`, write a program that adds them as numbers $1234 + 5678$, which should be 6912.

b. **Prime Numbers:** Write a program that computes all prime numbers smaller than a given number. Find out, how many prime numbers you can compute (don't output them in this test, this makes it slow) in one second.

c. **Faculty:** Compute the faculty with a recursive implementation. Use `long long` as your numeric type.

d. **Debug vs. Release**+**:**Compile the previous program for prime number finding in Release Mode (Visual Studio) or using flags like `gcc -o primes -std=c++11 -Ofast -march=native primes.c`. Compare the runtime of the debug version and the release version.

e. **Use your time*****:**Freely practice writing programs in C, let the MATLAB tasks be inspiration.