Department of Mathematics
Dr. Ulf Lorenz
Christian Brandenburg

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Introduction to Mathematical Software
5th Exercise Sheet

Exercise 1 (Hello, World!)

(a) Open a terminal and create the subdirectory `exercise5` in your home-directory.

(b) Change to the newly created subdirectory and start the xemacs editor with the command `xemacs &`.

(c) Make yourself familiar with the editor. The tutorial under `Help -> Tutorials -> Default` might help you. A reference card can be found at
http://www.digilife.be/quickreferences/QRC/XEmacs%20Reference%20Card.pdf.

(d) Write the `Hello, World!` program that has been presented in the lecture using the xemacs editor.

(e) An overview of the `%`-codes for `printf` and related routines can be found at
http://www.cppreference.com/wiki/c/io/printf.

(f) Compile and execute the program as shown in the lecture.

Exercise 2 (Greatest Common Divisor)

In the OWO Lecture in Linear Algebra, you learned *Euclid's Algorithm* for finding the greatest common divisor of two (positive) integers. Implement this algorithm in C/C++.

- Use your `Hello, World!` program as a starting point.

- To find out more on integer arithmatic, see
http://www.cplusplus.com/doc/tutorial/operators.html.
The *modulo* operator is '`%`', i.e. $a \mod b$ is implemented as `a % b`.

- Write a subroutine `int euclid (int a, int b){...}`

- you can implement the algorithm recursively, as defined in the lecture, or iteratively using a `while`-loop.

Exercise 3 (Number Conversion)

On Exercise Sheet 2, you learned an algorithm for converting numbers from decimal to binary representation. Implement this algorithm in `C/C++`. Use a `C-string` for the binary representation which you initialize as an empty string, i.e. `char s[100];` `strcpy(s,‘"’);`.

Hint: You can concatenate strings using `strcat`. Don't forget to `#include <string.h>` or `#include <cstring>`. You find more information on C-type strings and functions to manipulate them at http://www.cplusplus.com/reference/clibrary/cstring/.

Exercise 4 (Least Common Multiple)

Devise an algorithm to compute the least common multiple of two positive integers and implement it in `C/C++`.