

Computermathematik – Übung $\text{\LaTeX}2$

- **Abgabeschluss:** Di 28. 10. um 23:59
- **Präsentation:** Mi 29. 10. in der Übungsgruppe
- **Abgabeformat:** `.zip/.tex` (siehe unten), Deutsch oder Englisch

Aufgabe 4 – Dokument-Template (4 Punkte)

Erstelle eine wiederverwendbare Dateivorlage für Artikel (oder, alternativ, für Labor-Protokolle, Skripten oder Bachelor-Arbeiten). Diese Vorlage `dokument.tex` soll dabei hauptsächlich Präambel und Befehle z.B. für Titelseite, Inhaltsverzeichnis, Abbildungs- und Literaturverzeichnis und ähnliches enthalten. Inkludiere ein Struktur-Skelett mit wiederkehrenden Überschriften (z.B. Introduction, Experiment-Aufbau).

Demonstriere die Vorlage anhand eines Beispiels (beispielsweise Text und Bilder aus einem Wikipedia-Artikel). Verwende mindestens eine Literaturquelle, die in einem \LaTeX -Literaturverzeichnis eingebunden wird. Der \LaTeX -Eintrag kann entweder selbst geschrieben werden oder aus einer entsprechenden Online-Datenbank stammen (z.B. DBLP¹, Google Scholar² oder Homepage des Autors).

Abzugeben ist (ausnahmsweise!) eine `.zip`-Datei mit allen benötigten Dateien (`.tex`, `.bib` und Bilder, aber *nicht* `.pdf`, `.log`, `.aux`, ...).

Aufgabe 5 – Algorithmen (4 Punkte)

Implementiere einen einfachen Algorithmus in zwei Versionen:

1. in einer Programmiersprache (Umgebung z.B. `verbatim` oder `lstlisting`)
2. als Pseudocode (Umgebung z.B. `enumerate` oder `algorithmic`)

Abzugeben ist eine `.tex`-Datei, die beide Versionen enthält. Dabei soll die erste Version durch einen Parser/Interpreter korrekt ausführbar, die zweite hingegen auf die wesentlichen Teile reduziert und für den Leser besser verständlich sein. Die Programmiersprache für Version 1 ist beliebig, es darf auch Code aus anderen Lehrveranstaltungen o.ä. recyclet werden. Achte ggf. auf korrektes Syntax Highlighting. Version 2 darf sowohl eher formal als auch eher textuell sein, soll aber einfacher lesbar als Version 1 sein (und *nicht* recyclet, sondern selbst neu formuliert). Einige Vorschläge für simple Algorithmen:

- Turmrechnen (wie ESP Homework 2), *oder*
- Division mit Rest (Schulmethode oder wiederholt subtrahieren), *oder*
- Schnittmenge von zwei Mengen/Listen bilden, *oder*
- Rekursive Formel oder Folge bis zu einem bestimmten Glied berechnen, *oder* ...

Bonus – Typographische Feinheiten (2 Bonuspunkte)

Beachte in Aufgabe 4 typographische Feinheiten: korrekte Anführungszeichen (richtige Sprache!), Apostrophe, Gedanken- vs. Bindestriche, geschützte und halbe Leerzeichen, korrekte Worttrennung und so weiter. Abzugeben ist eine `.tex`-Datei, in der mindestens vier typographisch interessante Stellen aus Aufgabe 4 zitiert werden.

¹<http://www.informatik.uni-trier.de/~ley/db/>

²<http://scholar.google.at/>