



Fakultät für Mathematik
Lehrstuhl für Angewandte Geometrie und Diskrete Mathematik

Die tumthesis-Klasse

Eine Anleitung für Abschlussarbeiten

Kleiner Überblick von Wolfgang F. Riedl, Michael Ritter

Themensteller: Prof. Dr. Peter Gritzmann

Betreuer: Dr. René Brandenberg

Abgabedatum: 24. September 2014

Hiermit erkläre ich, dass ich diese Arbeit selbstständig angefertigt und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 24. September 2014

Wolfgang F. Riedl, Michael Ritter

Abstract

Here we give a short summary of the project or thesis of length at most a quarter of a page. This could be e. g. as follows:

This document is an introduction to the use of the \LaTeX -package `tumthesis.cls`, with which theses can be written in the TUM style. The basic structure of the example files is explained and some optional components are mentioned briefly. There are also some tips for \LaTeX beginners (and also for more advanced users who want to learn some more) as well as suggested reading for individual study.

Zusammenfassung

Hier schreibt man eine kurze Zusammenfassung der Arbeit im Umfang von maximal einer Viertelseite. Das kann z. B. so aussehen:

Die Arbeit führt in die Verwendung des \LaTeX -Pakets `tumthesis.cls` ein, mit dem Abschlussarbeiten im TUM-Stil gesetzt werden können. Die grundlegende Gliederung der Beispieldateien wird erklärt und auf optionale Bestandteile wird kurz eingegangen. Außerdem enthält der Text ein paar Tipps für \LaTeX -Anfänger (und auch für Fortgeschrittene, die noch etwas dazulernen wollen) sowie Literaturhinweise zum Selbststudium.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Dateien	1
1.2	Konfiguration und Optionen	2
1.3	Grundeinstellungen	3
1.4	Sprachauswahl und Zeichensatz	4
1.5	Druck	5
1.6	Titelseite	5
1.7	Wichtige Hinweise	5
1.7.1	Mathe-Umgebungen	5
1.7.2	BibLaTeX	5
1.8	Einige Pakete	6
1.8.1	Index	6
1.8.2	scrbook	7
1.8.3	csquotes	7
1.8.4	cleveref	8
1.8.5	ntheorem	8
1.8.6	booktabs	9
1.8.7	tabularx	9
1.8.8	TikZ	9
1.8.9	subcaption	10
1.8.10	fixme	10
1.8.11	hyperref	11
1.8.12	listings	11
1.8.13	algorithm2e	12
2	Zum Weiterlesen	13
A	Bemerkungen zur Implementierung	15
	Abbildungsverzeichnis	17
	Tabellenverzeichnis	19
	Index	19

Literatur	23
List of Corrections	25

Kapitel 1

Einleitung

Dieses Kapitel gibt ein kurzes Beispiel für die Verwendung der `tumthesis`-Klasse.

1.1 Dateien

Tabelle 1.1 zeigt alle zu diesem Beispiel gehörenden Dateien mit einer kurzen Erklärung.

Dateiname	Erklärung
<code>tumthesis.cls</code>	Klassendatei, stellt die grundlegenden Kommandos bereit und bindet wichtige Pakete ein
<code>tumcolors.sty</code>	L ^A T _E X-Paket, in dem die offiziellen Farben der TUM definiert werden, wird von <code>tumthesis.cls</code> benutzt
<code>thesis.tex</code>	Hauptdatei dieses Beispiels, Ausgangspunkt für die eigene Arbeit. Alle anderen <code>.tex</code> -Dateien werden von dieser Datei eingebunden.
<code>thesis.pdf</code>	PDF-Version von <code>thesis.tex</code>
<code>preamble.tex</code>	Präambel, in welcher vom Nutzer Pakete geladen werden können
<code>abstract.tex</code>	Text für die Abstracts in Deutsch und Englisch
<code>introduction.tex</code>	Text für dieses Kapitel
<code>conclusion.tex</code>	Text für das folgende Kapitel
<code>appendix.tex</code>	Text für den Anhang
<code>thesis.bib</code>	Bib _T E _X -Datei für die Literaturhinweise
<code>TUMlMblau</code>	Logo der Mathematik-Fakultät für die Titelseite
<code>TUMloblauSchriftzugL</code>	TUM-Logo für die Titelseite
<code>TopMath-Bildmotiv.jpg</code>	TopMath Logo für die Titelseite

Tabelle 1.1: zum Beispiel gehörige Dateien

Um aus den Quelldateien das fertige PDF zu erstellen, sind folgende Befehle aufzurufen:

```
pdflatex thesis
biber thesis
makeindex -s myindex.ist
pdflatex thesis
pdflatex thesis
```

Der erste Durchlauf von `pdflatex` erzeugt einige Hilfsdateien und eine (fast schon fertige) PDF-Ausgabe – lediglich einige Grafiken können noch an der falschen Stelle sitzen und die Verweise und Literaturangaben funktionieren noch nicht. Mit dem `biber`-Aufruf arbeitet das System dann die Datei `thesis.bib` durch und erzeugt daraus das Literaturverzeichnis (siehe Abschnitt 1.7.2). Anschließend wird mit `makeindex` noch der Index erstellt (siehe Abschnitt 1.8.1). Die beiden folgenden Aufrufe von `pdflatex` dienen dazu, die Verweise korrekt zu setzen und die Platzierung der Grafiken auszurichten.

Fortgeschrittene können den Prozess übrigens auch automatisieren, indem einfach

```
latexmk --pdf thesis
```

aufgerufen wird. Das Tool `latexmk` sorgt dann automatisch für die korrekte Anzahl von Aufrufen und für den Bib_TE_X-Durchlauf.

1.2 Konfiguration und Optionen

Eigene Pakete des Nutzers sowie Änderung von vorgenommenen Einstellungen können in der Datei `preamble.tex` geladen werden. Der dort enthaltene Code wird zum Ende der Klasse (unmittelbar vor dem Laden der letzten Pakete `hyperref` und `cleveref`) eingelesen und ausgeführt.

Die `tumthesis`-Klasse akzeptiert einige Optionen, die das Aussehen der Titelseite und anderes Verhalten ändern:

- `topmath`: Diese Option platziert das TopMath-Bildmotiv auf der Titelseite:

```
\documentclass[topmath]{tumthesis}
```

Alternativ kann man auch ein eigenes Motiv auf der Titelseite platzieren:

- `titlepicture`: Name der einzubindenden Datei
- `titlepictureX`: Horizontaler Abstand (mit Einheit) zwischen rechter unteren Ecke der Seite und rechter unteren Ecke des Bildes
- `titlepictureY`: Vertikaler Abstand (mit Einheit) zwischen rechter unteren Ecke der Seite und rechter unteren Ecke des Bildes

```
\documentclass[titlepicture=MA_CMYK.pdf,titlepictureX=25mm,
titlepictureY=40mm]{tumthesis}
```

würde nochmals das Mathelogo rechts unten auf der Seite einfügen.

Weiterhin kann auch das Verhalten des Inhalts von Theoremen angepasst werden:

- **theoremtitle**: Soll der Inhalt eines Theorems neben dem Titel begonnen werden (nobreak) oder in der nächsten Zeile (break, Standardoption)

```
\documentclass[theoremtitle=nobreak]{tumthesis}
```

Eine weitere Option erlaubt die Konfiguration des BibLaTeX-Backends:

- **biblatexBackend**: Die Defaulteinstellung ist Biber (Parameter biber), alternativ sind alle Optionen des Parameters „backend“ des BibLaTeX-Packets möglich:

```
\documentclass[biblatexBackend=bibtex]{tumthesis}
```

1.3 Grundeinstellungen

Ganz zu Beginn werden in der Datei `thesis.tex` ein paar wichtige Einstellungen vorgenommen. Der Code sieht wie folgt aus:

```
% -----
% PDF-Information
\hypersetup{
  pdfauthor={Wolfgang Ferdinand Riedl, Michael Ritter},
5  pdftitle={Die tumthesis-Klasse},
  pdfsubject={Anleitung für Abschlussarbeiten},
  pdfkeywords={Masterarbeit, Bachelorarbeit}
  colorlinks=true, %farbige Links (für die PDF-Version)
% colorlinks=false, % keine farbigen Links (für die ↪
  ↪ Druckversion)
10 }

% -----

% Basisdaten
15 \author{Wolfgang F. Riedl, Michael Ritter}
  \title{Die \texttt{tumthesis}-Klasse}
  \subtitle{Eine Anleitung für Abschlussarbeiten}
  \faculty{Fakultät für Mathematik}
```

```
20 \institute{Lehrstuhl für Angewandte Geometrie und Diskrete ~
    ↪ Mathematik}
    %\subject{master}
    %\subject{bachelor}
    %\subject{diploma}
    %\subject{project}
25 %\subject{seminar}
    %\subject{idp}
    \subject{Kleiner Überblick}
    \professor{Prof. Dr. Peter Gritzmann} %Themensteller
    \advisor{Dr. René Brandenberg} %Betreuer
30 \date{26.12.2012} %Abgabedatum
    \place{München} %Ort für die Unterschrift
```

Die Angaben im `hypersetup`-Befehl erscheinen zwar nicht im Dokument selbst, sie werden aber als Metadaten in die PDF-Datei eingebettet und lassen sich im Acrobat Reader (und vielen anderen PDF-Betrachtern) anzeigen. Bei den restlichen Befehlen dürfte das meiste selbsterklärend sein. Beim `subject{}` kann man entweder einen beliebigen Text angeben (wie das in diesem Beispiel mit „Kleiner Überblick“ gemacht wird) oder man verwendet eines der vordefinierten Schlüsselwörter, die automatisch die Ausgabe „Masterarbeit“, „Bachelorarbeit“ bzw. eine passende andere Ausgabe erzeugen. Dazu entfernt man einfach das Kommentarsymbol in der betreffenden Zeile und kommentiert dafür den jetzt aktiven `subject`-Befehl aus. Es ist natürlich darauf zu achten, dass man die Bezeichnungen der gewählten Sprache anpasst, vgl. dazu Abschnitt 1.4. (Man kann den `\subject{}`-Befehl übrigens auch ganz weglassen, dann wird eben keine Bezeichnung auf der Titelseite erzeugt und nur der Autor genannt.)

1.4 Sprachauswahl und Zeichensatz

Die Klasse unterstützt als Sprachen Deutsch und Englisch. Zu Beginn von `thesis.tex` gibt es den Befehl

```
\selectlanguage{ngerman}
```

mit dem die Grundsprache eingestellt wird. Diese kann man mit genau diesem Befehl bzw. `\selectlanguage{english}` übrigens jederzeit im Dokument ändern. Dabei passen sich ein paar Einstellungen automatisch an, z. B. liefern die Befehle `\eg` und `\ie` jeweils den passenden Text (die sollte man übrigens verwenden, weil sie gleich für den typographisch korrekten Abstand sorgen) und die Überschriften ändern sich, aber auch etwas subtilere Dinge wie die verwendeten Trennmuster für die automatische Silbentrennung. Ein Beispiel für so eine Umschaltung findet man im Abstract.

Für eigene Dateien ist es wichtig, das korrekte „Encoding“ zu wählen. Vorgabe ist hier Unicode (UTF-8). Das ermöglicht es, Umlaute und andere Sonderzeichen direkt einzugeben, erfordert aber unter Umständen eine richtige Einstellung im Editor.

Besonders unter Windows-Systemen sind manche Editoren standardmäßig auf Latin-1 statt auf Unicode eingestellt – so etwas kann zu sehr merkwürdigen Fehlermeldungen führen!

1.5 Druck

Beim Drucken ist unbedingt darauf zu achten, dass die Arbeit doppelseitig gedruckt wird. Die Seitenränder und die Kopf- und Fußzeilen sind darauf ausgelegt, dass die Arbeit doppelseitig gedruckt und dann gebunden wird.

Um die Bindekorrektur anzupassen (und so zum Beispiel in der Mitte mehr Platz für die Bindung zu lassen) muss einfach die Zeile

```
BCOR =5 mm % Binding correction , ensures sufficient space ↷  
↪ for binding
```

in der Datei `tumthesis.cls` angepasst werden.

1.6 Titelseite

Standardmäßig wird eine dem TUM-Styleguide weitestmöglich folgende Titelseite dargestellt. Durch das Ersetzen der Zeile

```
\maketitlepage%
```

in der Datei `thesis.tex` durch

```
\maketitlepageDissertation%
```

wird diese durch eine an eine Dissertation angepasste Titelseite ersetzt.

1.7 Wichtige Hinweise

1.7.1 Mathe-Umgebungen

Da diese Klasse das Paket `ntheorem` lädt, werden durch Matheumgebungen der Art `\[... \]` Fehlermeldungen erzeugt. Solche Umgebungen sollten stattdessen durch `\begin{equation*} ... \end{equation*}` ersetzt werden.

1.7.2 BibLaTeX

Das Paket nutzt als Standard das Biber Backend, welches UTF-8 kodierte Bibliographiedateien lesen kann. Die Umstellung auf ein anderes Backend erfolgt über den Parameter `biblatexBackend` (siehe Parameter).

Für die meisten \LaTeX -Editoren findet man im Web Anleitungen für die Einrichtung mit biber.

1.8 Einige Pakete

Die Klasse `tumthesis.cls` bindet bereits eine ganze Reihe nützlicher Pakete ein, die wir hier kurz auflisten wollen.

1.8.1 Index

Die `tumthesis` Klasse lädt das Paket `imakeidx`, welches die schnelle und einfache Erstellung eines *Index* erlaubt. Um ein Wort zum Index hinzuzufügen muss einfach der Befehl `\index{Schlüsselwort}` angehängt werden. Das Wort „Index“ wird zum Beispiel wie folgt zum Index dieses Dokuments hinzugefügt:

```
... einfach Erstellung eines \emph{Index}\index{Index} ~
    ↪ erlaubt. Um ...
```

Symbole können ebenso einfach hinzugefügt werden: Das Symbol ζ wird mit folgender Zeile zum Index hinzugefügt:

```
... Das Symbol $\zeta$\index{$\zeta$} wird mit ...
```

Um die Position eines Symbols (oder anderen Elements) im Index zu ändern, kann man ein zusätzliches Schlüsselwort (welches auch eine Formel sein kann) angeben:

Um das Symbol π so in den Index aufzunehmen, dass es an der Stelle des Wortes „pi“ und zusätzlich noch an der Position des Symbols „ p_i “ erscheinen würde, kann man den folgenden Code nutzen:

```
... as Symbol $\pi$\index{pi@$pi$}\index{$p_i$@$pi$} so in ~
    ↪ den ...
```

Subkategorien können auch sehr einfach über `\index{Schlüsselwort!Subkategorie}` erstellt werden: Die Definition eines metrischen Raums

Definition 1.1 (Metric)

Let X be a set and $d : X \times X \rightarrow \mathbb{R}$. The function d is a metric on X if the following three properties hold for all $x, y, z \in X$

1. $d(x, x) \geq 0$ and $d(x, y) = 0 \iff x = y$ (non-negativity)
2. $d(x, y) = d(y, x)$ (symmetry)
3. $d(x, z) \leq d(x, y) + d(y, z)$ (triangle inequality).

kann beispielsweise im Index mit Hilfe des folgenden Codes referenziert werden:

```
\begin{definition}[Metric]
  \index{metric}
  Let  $X$  be a set and  $d: X \times X \rightarrow \mathbb{R}$ 
    ↪  $\mathbb{R}$ . The function  $d$  is a metric on  $X$  if
    ↪ the following three properties hold for all  $x, y, z \in$ 
    ↪  $X$ 
```

```

\begin{enumerate}
5 \item  $d(x,x) \geq 0$  and  $d(x,y) = 0$  iff  $x = y$   $\leadsto$ 
     $\hookrightarrow$  (non-negativity)\index{metric!non-negativity}
    \item  $d(x,y) = d(y,x)$  (symmetry)\index{metric!symmetry}
    \item  $d(x,z) \leq d(x,y) + d(y,z)$  (triangle  $\leadsto$ 
         $\hookrightarrow$  inequality)\index{metric!triangle inequality}.
\end{enumerate}
\end{definition}

```

Zur Erstellung des Index muss die folgende Zeile

```
\makeindex[title=Index,options=-s myindex]
```

zur Datei `thesis.tex` vor dem Befehl `\begin{document}` hinzugefügt werden! Die Option `title=Index` gibt die Überschrift des Index (in diesem Fall „Index“) an; die zweite Option eine angepasste Style-Datei (funktioniert nicht in allen Umgebungen, andernfalls müssen die unten aufgeführten Kommandozeilenoptionen genutzt werden).

Der Index kann dann über das folgende Kommando zum Dokument hinzugefügt werden:

```

%Add the index to the table of contents
\addcontentsline{toc}{chapter}{Index}
%print the index
\printindex

```

Der Index muss mit Hilfe des `makeindex`-Befehl kompiliert werden (was die meisten Editoren automatisch machen). Das Layout des Index kann mit Hilfe einer modifizierten Style-Datei angepasst werden, diese wird über die Option `-s stylefile.ist` geladen (oder unter Umständen über die oben dem `\makeindex`-Befehl übergebene Option). Dieses Dokument wurde mit Hilfe des in der beiliegenden Datei `dmyindex.ist` definierten Styles erstellt.

1.8.2 scrbook

Die Klasse `tumthesis.cls` baut komplett auf `scrbook.cls` auf. Das bedeutet insbesondere, dass auch alle Optionen und Befehle von `scrbook` zur Verfügung stehen. Genauer dazu findet man in der Dokumentation [KM12a] oder in gedruckten Ausgabe Kohm und Morawski [KM12b].

1.8.3 csquotes

Dieses Paket liefert unter anderem den Befehl `\enquote{}`, mit dem sich automatisch korrekte Anführungszeichen setzen lassen. Dabei richtet sich das Paket nach der gerade aktiven Sprache: In deutschen Texten erscheinen „deutsche Anführungszeichen“, while English texts use “corresponding quotation marks”.

1.8.4 cleveref

Verweise unter \LaTeX setzt man normalerweise mit `\ref{}`. Dieses Paket definiert die neuen Befehle `\cref` und `\Cref`, die dafür sorgen, dass neben der richtigen Nummer automatisch auch ein beschreibender Text gesetzt wird (und zwar in der gerade eingestellten Sprache). Die zweite Version sorgt dabei für Großschreibung und sollte daher am Satzanfang verwendet werden (auch wenn das im Deutschen meistens keinen Unterschied macht, weil die Bezeichnungen häufig Substantive sind, die ohnehin großgeschrieben werden). Ein Beispiel kann man weiter oben und auch hier sehen: Die Verweise auf Tabelle 1.1 werden mit `cleveref` erzeugt, der Text „Tabelle“ wird dabei automatisch eingefügt.

1.8.5 ntheorem

Mit diesem Paket werden eine Reihe Standard-Umgebungen für Definitionen, Sätze, Beweise etc. bereitgestellt. Die Bezeichnungen sind übrigens sprachabhängig. Ein Beispiel:

Definition 1.2

Jedes Element eines Vektorraums bezeichnen wir als *Vektor*.

Satz 1.3 (Hauptsatz der kanonischen Vektorraumbezeichnungen)

Zu jedem Vektor v gibt es einen Vektorraum V mit $v \in V$.

Beweis. Der triviale Beweis wird dem Leser zu Übung überlassen. Es ist wirklich nicht schwer, probieren Sie es einfach. \square

Let us now demonstrate an English version of the above proof:

Proof. The proof of Theorem 1.3 is most trivial and only complete idiots would not be able to do it themselves. If you even bothered reading this proof you might want to think about studying some other subject. \square

Wir formulieren noch einen Satz, um nebenbei ein weiteres Feature von `cref` vorzuführen, mit dem man nämlich auch mehrere Verweise zusammenpacken kann, hierzu sei auf Sätze 1.3 und 1.4 verwiesen.

Satz 1.4

\LaTeX ist toll!

Die Umgebungen lassen sich natürlich auch ergänzen und dem eigenen Geschmack anpassen. Dazu kann entweder die Beispiele in `tumthesis.cls` anschauen oder man zieht die Dokumentation [MS11] zum `ntheorem`-Paket zu Rate. In Tabelle 1.2 sind alle vordefinierten Umgebungen aufgelistet.

Umgebung	Text	
	Deutsch	Englisch
<code>definition</code>	Definition	Definition
<code>theorem</code>	Satz	Theorem
<code>satz</code>	Satz	Theorem
<code>lemma</code>	Lemma	Lemma
<code>proposition</code>	Proposition	Proposition
<code>corollary</code>	Korollar	Corollary
<code>korollar</code>	Korollar	Corollary
<code>remark</code>	Bemerkung	Remark
<code>bemerkung</code>	Bemerkung	Remark
<code>example</code>	Beispiel	Example
<code>beispiel</code>	Beispiel	Example
<code>proof</code>	Beweis	Proof
<code>beweis</code>	Beweis	Proof
<code>conjecture</code>	Vermutung	Conjecture
<code>vermutung</code>	Vermutung	Conjecture
<code>problem</code>	Problem	problem

Tabelle 1.2: vordefinierte `ntheorem`-Umgebungen

1.8.6 booktabs

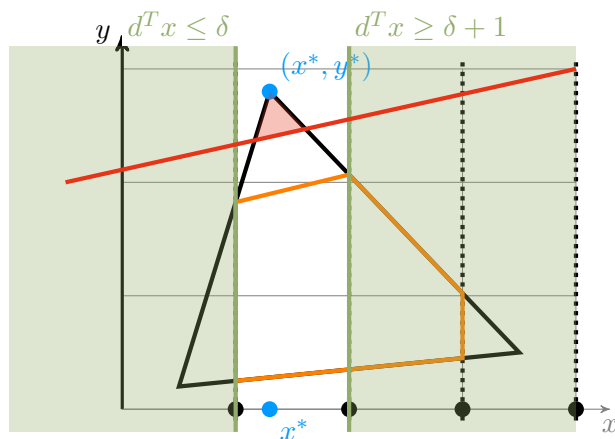
Mit diesem Paket lassen sich schönere Tabellen setzen, Tabelle 1.1 zeigt ein Beispiel. Viele Hinweise zum Tabellensatz finden sich auch in der ausführlichen Dokumentation zu diesem Paket.

1.8.7 tabularx

Tabellen, welche die Breite bestimmter Spalten ändern, um eine vorgegebene Breite zu haben, können mit Hilfe von `tabularx` erzeugt werden.

1.8.8 TikZ

TikZ ist zwar kein Zeichenprogramm, man kann damit aber trotzdem verdammt gute Abbildungen erzeugen. Die Dokumentation [Tan08] ist sehr ausführlich, im Internet findet man unter <http://www.texample.net> eine Menge Beispiele, die zeigen, was mit dem Paket alles möglich ist. Damit das Abbildungsverzeichnis nicht so leer bleibt, fügen wir hier in Abb. 1.1a mal ein TikZ-Bild ein. Keine Angst, den Code müssen Sie nicht direkt verstehen. In der TikZ-Anleitung gibt es auch ein paar schön erklärte, einfachere Beispiele.



(a) Beispiel einer Split Disjunction



(b) Das TUM Logo

Abbildung 1.1: Zwei Graphiken

1.8.9 subcaption

Um innerhalb einer Figure mehrere Subfigures zu erzeugen, kann man die folgende Umgebung aus dem Paket `subcaption` nutzen: `\begin{subfigure} ... \end{subfigure}`. Hiermit kann man Subfigure und Subtables mit einer zu figures und tables identischen Syntax erzeugen.

1.8.10 fixme

FiXme Anm!

Mit diesem Paket kann man Anmerkungen im Dokument erstellen, die noch nötige Arbeiten dokumentieren. Ganz am Ende des Dokuments gibt es dann eine „List of Corrections“, in der alle Notizen aufgelistet sind. In diesem Absatz sind zur Demonstration zwei solche FixMe-Hinweise eingebaut – man sieht das einmal an den Anmerkungen am Rand, andererseits aber auch ganz hinten in diesem Dokument in der erwähnten „List of Corrections“. Auch hier kann man zahlreiche Einstellungen vornehmen, die *Dokumentation* ist empfehlenswert. Erwähnt sei eine Einstellung, die man ganz am Anfang der Datei `thesis.tex` vornehmen sollte:

FiXme Anm!

```
%FixMe-Status: final (keine FixMe-Anmerkungen) oder draft ↷
                  ↷ (Anmerkungen sichtbar)
\fixsetup{draft}
%\fixsetup{final}
```

Ersetzt man hier die „draft“-Zeile durch die „final“-Zeile, so passieren zwei Dinge: Alle `\fixfatal{}`-Kommandos werden zu \LaTeX -Fehlern, der \TeX -Lauf bricht bei so einem Kommando also ab (nützlich, um wirklich böse Fehler anzumerken, die man auf keinen

Fall übersehen darf). Alle anderen fixme-Kommandos (also `\fxnote{}`, `\fxwarning{}`, `\fxerror{}`) werden dagegen unsichtbar, die Markierungen im Text und auch die „List of Corrections“ am Ende des Dokuments verschwinden. Genauer zu den möglichen Kommandos und zahlreichen Einstellmöglichkeiten bietet die Dokumentation [Ver09].

1.8.11 hyperref

Mit dem `hyperref`-Paket werden einige PDF-Einstellungen vorgenommen (vgl. Abschnitt 1.3). Außerdem sorgt das Paket dafür, dass alle Referenzen, Literaturverweise und das Inhaltsverzeichnis zu klickbaren Links werden, mit denen man im Dokument hin- und herspringen kann. Standardmäßig werden diese Links auch in schwarzer Farbe gesetzt, sie sind also nicht sofort sichtbar. Alternativ kann man mit der Einstellung `colorlinks=true` im `\hypersetup{}`-Befehl zu Beginn des Dokuments aber auch dafür sorgen, dass Links in einem dezenten dunkelblauen Farbton erscheinen. Für die Bildschirm-Version ist das praktisch, für die Druckversion sollte man aber auf schwarze Farbe zurückschalten, also die Option `colorlinks=false` verwenden (schließlich kann man Links im Ausdruck ja nicht anklicken).

```
\hypersetup{
  pdfauthor={Wolfgang Ferdinand Riedl, Michael Ritter},
  pdftitle={Die tumthesis-Klasse},
  pdfsubject={Anleitung für Abschlussarbeiten},
5 pdfkeywords={Masterarbeit, Bachelorarbeit},
  colorlinks=true, %farbige Links (für die PDF-Version)
  % colorlinks=false, % keine farbigen Links (für die ↪
    ↪ Druckversion)
}
```

1.8.12 listings

Mit dem Paket `listings` kann man Quellcode-Listings schön formatiert darstellen. In diesem Beispiel wird es benutzt, um \LaTeX -Quellcode darzustellen. Mit den Standardeinstellungen in diesem Paket sorgt es automatisch für Zeilennummerierung, Zeilenumbruch und einige anderen Kleinigkeiten. Natürlich lässt sich vieles individuell anpassen, Details dazu sind in der Dokumentation [HM07] zu finden. Eine kleine Warnung: Das Paket ist so eingestellt, dass es mit Umlauten und „ß“ in Quelltexten zurechtkommt, andere Sonderzeichen können aber Probleme machen (auch in Kommentaren). Am besten, man vermeidet Sonderzeichen in Quelltexten komplett – wenn es aber nicht anders geht, kann man sich in `tumthesis.cls` anschauen, wie die Einstellungen ergänzt werden müssen, um auch andere Sonderzeichen zu behandeln.

1.8.13 algorithm2e

Mit `algorithm2e` hat man eine weitere Möglichkeit, Quelltext darzustellen. Im Gegensatz zu `listings` wird hier jedoch nicht Code für eine bestimmte Programmiersprache formatiert, vielmehr wird formatierter Pseudocode mit einer eigenen leicht verständlichen Syntax erzeugt.

Kapitel 2

Zum Weiterlesen

In diesem Kapitel sammeln wir ein paar Literaturhinweise in Sachen \LaTeX , die Anfängern und Fortgeschrittenen Anwendern den Einstieg erleichtern und vielleicht einige nützliche Hinweise geben können.

lshort: „The Not So Short Introduction to \LaTeX “ (vgl. [Oet+11]) ist eine aktuelle Einführung, die sich mit moderatem Zeitaufwand zum Einstieg durcharbeiten lässt (die Autoren geben für die zum Zeitpunkt der Erstellung dieses Dokuments aktuelle Version 5.01 eine Zeit von 157 Minuten an). Die jeweils aktuelle Version findet man auf <http://tobi.oetiker.ch/lshort/lshort.pdf>.

\LaTeX and Friends: Das Buch [Don12] ist eine empfehlenswerte und aktuelle Einführung in \LaTeX , die viele aktuelle Pakete behandelt. Für Einsteiger wie auch für Fortgeschrittene einen Blick wert.

l2tabu: In [ET11] sind einige Hinweise auf veraltete Pakete und \LaTeX -Befehle enthalten, die besonders für Fortgeschrittene \LaTeX -Nutzer lesenswert sind. Hier erfährt man, warum man bestimmte Kommandos besser nicht einsetzen sollte und was bessere Alternativen sind. Übrigens: Die Klasse `tumthesis.cls` bindet automatisch das Paket `nag` ein, das bei vielen in l2tabu gelisteten Fehler direkt Alarm schlägt.

Anhang A

Bemerkungen zur Implementierung

Im Anhang können beispielsweise Code-Listings oder weiterführende Bemerkungen ausgeführt werden, die im Hauptteil den Lesefluss stören würden. Wenn Sie keinen Anhang benötigen, können Sie diese Datei natürlich auch einfach weglassen (und dann auch den `\include`-Befehl in `thesis.tex` löschen).

Abbildungsverzeichnis

1.1	Zwei Graphiken	10
-----	--------------------------	----

Alle Abbildungen in diesem Dokument stammen vom Autor selbst. Zur Erzeugung wurde das großartige \TeX -Paket TikZ von Till Tantau verwendet, vgl. [[Tan08](#)].

Tabellenverzeichnis

1.1	zum Beispiel gehörige Dateien	1
1.2	vordefinierte <code>ntheorem</code> -Umgebungen	9

Index

Symbole	_____
ζ , 6	
π , 6	
I	_____
Index, 6	
index	
<code>imakeidx</code> , 6	

M	_____
metric, 6	
non-negativity, 6	
symmetry, 6	
triangle inequality, 6	
P	_____
π , 6	

Literatur

- [Don12] M. R. C. van Dongeren. *LaTeX and Friends*. Springer, 2012.
- [ET11] M. Ensenbach und M. Trettin. *Das LATEX 2e-Sündenregister*. 2011.
- [HM07] C. Heinz und B. Moses. *The Listings Package*. 2007.
- [KM12a] M. Kohm und J.-U. Morawski. *KOMA-Script*. 2012.
- [KM12b] M. Kohm und J.-U. Morawski. *KOMA-Script*. 4. Auflage. Berlin: Verlag:Lehmanns, 2012.
- [MS11] W. May und A. Schedler. *An Extension of the LATEX-Theorem Environment*. 2011.
- [Oet+11] T. Oetiker, H. Partl, I. Hyna und E. Schlegl. *The Not So Short Introduction to LATEX2e*. 2011.
- [Tan08] T. Tantau. *The TikZ and PGF Packages. Manual for version 2.00*. Lübeck, Feb. 2008.
- [Ver09] D. Verna. *FiXme – Collaborative annotation tool for LATEX*. 2009.

List of Corrections

Anm: Hier könnte noch ein Beispiel folgen.	10
Anm: ins Literaturverzeichnis aufnehmen	10