LaTeX 2_ε-Kurzbeschreibung

Version 2.1
18. April 1999

Walter Schmidt*
Jörg Knappen†
Hubert Partl‡
Irene Hyna§

LaTeX ist ein Satzsystem, das für viele Arten von Schriftstücken verwendet werden kann, von einfachen Briefen bis zu kompletten Büchern. Besonders geeignet ist es für wissenschaftliche oder technische Dokumente. LaTeX ist für praktisch alle verbreiteten Betriebssysteme verfügbar.
Die vorliegende Kurzbeschreibung bezieht sich auf die Version LaTeX 2_ε in der Fassung vom 1. Juni 1998 und sollte für den Einstieg in LaTeX ausreichen. Eine vollständige Beschreibung enthält das *LaTeX-Handbuch* [1] in Verbindung mit der Online-Dokumentation.

*Erlangen, `<walter.schmidt@arcormail.de>`
†Electronic Technologies, Springer-Verlag, Heidelberg
‡Zentraler Informatikdienst der Universität für Bodenkultur, Wien
§Bundesministerium für Wissenschaft und Verkehr, Wien

Dieses Dokument wurde mit LaTeX gesetzt. Sein Quelltext ist zu finden unter `<ftp://ftp.dante.de/tex-archive/info/lshort/german/>`.

Die Autoren danken Michael Hofmann, Rainer Schöpf, Stefan Steffens, Luzia Dietsche, Bernd Raichle und Heiko Oberdiek für Tips, Anmerkungen und Korrekturen.

Inhaltsverzeichnis

1 Allgemeines

1.1 The Name of the Game

1.1.1 TEX

TEX (sprich „Tech“, kann auch „TeX“ geschrieben werden) ist ein Computer-progamm von Donald E. Knuth [14, 15]. Es dient zum Setzen und Drucken von Texten und mathematischen Formeln.

1.1.2 LATEX

LATEX (sprich „Lah-tech“ oder „Lej-tech“, kann auch „LaTeX“ geschrieben werden) ist ein auf TEX aufbauendes Computerprogramm und wurde von Leslie Lamport [1, 3] geschrieben. Es vereinfacht den Umgang mit TEX, indem es entsprechend der logischen Struktur des Dokuments auf vorgefertigte Layout-Elemente zurückgreift.

1.1.3 LATEX 2ε

LATEX 2ε (sprich „LATEX zwei e“) ist die aktuelle Variante von LATEX seit dem 1. Juni 1994. (Die vorherige hieß LATEX 2.09.) Wenn hier von LATEX gesprochen wird, so ist normalerweise dieses LATEX 2ε gemeint.

Neue Versionen von LATEX 2ε (z. B. mit Fehlerberichtigungen oder Ergänzungen) erscheinen zweimal jährlich im Juni und im Dezember; die vorliegende Beschreibung setzt mindestens diejenige vom Juni 1998 voraus.

1.2 Grundkonzept

1.2.1 Autor, Designer und Setzer

Für eine Publikation übergab der Autor dem Verleger traditionell ein maschinengeschriebenes Manuskript. Der Buch-Designer des Verlages entschied dann über das Layout des Schriftstücks (Länge einer Zeile, Schriftart, Abstände vor und nach Kapiteln usw.) und schrieb dem Setzer die dafür notwendigen Anweisungen dazu. LATEX ist in diesem Sinne der Buch-Designer, das Programm TEX ist sein Setzer.

Ein menschlicher Buch-Designer erkennt die Absichten des Autors (z. B. Kapitel-Überschriften, Zitate, Beispiele, Formeln . . .) meistens aufgrund seines Fachwissens aus dem Inhalt des Manuskripts. LATEX dagegen ist „nur“ ein Programm und benötigt daher zusätzliche Informationen vom Autor, die die logische Struktur des Textes beschreiben. Diese Informationen werden in Form von sogenannten „Befehlen“ innerhalb des Textes angegeben. Der Autor braucht sich also (weitgehend) nur um die logische Struktur seines Werkes zu kümmern, nicht um die Details von Gestaltung und Satz.

Im Gegensatz dazu steht der visuell orientierte Entwurf eines Schriftstückes mit Textverarbeitungs- oder DTP-Programmen wie z. B. WORD. In diesem Fall legt der Autor das Layout des Textes gleich bei der interaktiven Eingabe fest. Dabei sieht er am Bildschirm das, was auch auf der gedruckten Seite stehen wird. Solche Systeme, die das visuelle Entwerfen unterstützen, werden auch WYSIWYG-Systeme („what you see is what you get“) genannt.

Bei LaTeX sieht der Autor beim Schreiben des Eingabefiles in der Regel noch nicht sofort, wie der Text nach dem Formatieren aussehen wird. Er kann aber jederzeit einen „Probe-Ausdruck“ seines Schriftstücks auf dem Bildschirm machen und danach sein Eingabefile entsprechend korrigieren und die Arbeit fortsetzen.

1.2.2 Layout-Design

Typographisches Design ist ein Handwerk, das erlernt werden muß. Ungeübte Autoren machen oft gravierende Formatierungsfehler. Fälschlicherweise glauben viele Laien, daß Textdesign vor allem eine Frage der Ästhetik ist – wenn das Schriftstück vom künstlerischen Standpunkt aus „schön“ aussieht, dann ist es schon gut „designed“. Da Schriftstücke jedoch gelesen und nicht in einem Museum aufgehängt werden, sind die leichtere Lesbarkeit und bessere Verständlichkeit wichtiger als das schöne Aussehen.

Beispiele: Die Schriftgröße und Numerierung von Überschriften soll so gewählt werden, daß die Struktur der Kapitel und Unterkapitel klar erkennbar ist. Die Zeilenlänge soll so gewählt werden, daß anstrengende Augenbewegungen des Lesers vermieden werden, nicht so, daß der Text das Papier möglichst schön ausfüllt.

Mit interaktiven visuellen Entwurfssystemen ist es leicht, Schriftstücke zu erzeugen, die zwar „gut“ aussehen, aber ihren Inhalt und dessen Aufbau nur mangelhaft wiedergeben. LaTeX verhindert solche Formatierungsfehler, indem es den Autor dazu zwingt, die logische Struktur des Textes anzugeben, und dann automatisch ein dafür geeignetes Layout verwendet.

Daraus ergibt sich, daß LaTeX insbesondere für Dokumente geeignet ist, wo vorgegebene Gestaltungsprinzipien auf sich wiederholende logische Textstrukturen angewandt werden sollen. Für das – notwendigerweise – visuell orientierte Gestalten etwa eines Plakates ist LaTeX hingegen aufgrund seiner Arbeitsweise weniger geeignet.

1.2.3 Vor- und Nachteile

Gegenüber anderen Textverarbeitungs- oder DTP-Programmen zeichnet sich LaTeX vor allem durch die folgenden Vorteile aus:

- Der Anwender muß nur wenige, leicht verständliche Befehle angeben, die die logische Struktur des Schriftstücks betreffen, und braucht sich um die gestalterischen Details (fast) nicht zu kümmern.
- Das Setzen von mathematischen Formeln ist besonders gut unterstützt.
- Auch anspruchsvolle Strukturen wie Fußnoten, Literaturverzeichnisse, Tabellen u. v. a. können mit wenig Aufwand erzeugt werden.
- Routineaufgaben wie das Aktualisieren von Querverweisen oder die Erstellung des Inhaltsverzeichnisses werden automatisch erledigt.

- Es stehen zahlreiche vordefinierte Layouts zur Verfügung.
- LaTeX-Dokumente sind zwischen verschiedenen Installationen und Rechnerplattformen austauschbar.
- Im Gegensatz zu den meisten WYSIWIG-Programmen ist LaTeX auch in Verbindung mit langen oder komplexen Dokumenten stabil, und sein Ressourcenverbrauch (Speicher, Rechenzeit) ist geringer.

Ein Nachteil soll freilich auch nicht verschwiegen werden:

- Innerhalb der von LaTeX unterstützten Dokument-Layouts können zwar einzelne Parameter leicht variiert werden, grundlegende Abweichungen von den Vorgaben sind aber nur mit größerem Aufwand möglich (Design einer neuen Dokumentklasse, siehe [10, 17, 18, 19]).

1.2.4 Der Arbeitsablauf

Der typische Ablauf beim Arbeiten mit LaTeX ist:

1. Ein Eingabefile schreiben, das den Text und die LaTeX-Befehle enthält.
2. Dieses File mit LaTeX bearbeiten; dabei wird eine Datei erzeugt, die den gesetzten Text in einem geräteunabhängigen Format (dvi) enthält.
3. Einen „Probeausdruck" davon auf dem Bildschirm anzeigen (Preview).
4. Wenn nötig, die Eingabe korrigieren und zurück zu Schritt 2.
5. Das dvi-File ausdrucken.

Zeitgemäße Betriebssysteme machen es möglich, daß der Texteditor und das Preview-Programm gleichzeitig in verschiedenen Fenstern „geöffnet" sind; beim Durchlaufen des obigen Zyklus brauchen sie also nicht immer wieder von neuem gestartet werden. Nur die wiederholte LaTeX-Bearbeitung des Textes muß noch von Hand angestoßen werden und läuft ebenfalls in einem eigenen Fenster ab.

Wie man auf die einzelnen Programme – Editor, LaTeX, Previewer, Druckertreiber – in einer bestimmten Betriebssystemumgebung zugreift, sollte im *Local Guide* [7] beschrieben sein.

2 Eingabefile

Das Eingabefile für LaTeX ist ein Textfile. Es wird mit einem Editor erstellt und enthält sowohl den Text, der gedruckt werden soll, als auch die Befehle, aus denen LaTeX erfährt, wie der Text gesetzt werden soll.

2.1 Leerstellen

„Unsichtbare" Zeichen wie das Leerzeichen, Tabulatoren und das Zeilenende werden von LaTeX einheitlich als Leerzeichen behandelt. *Mehrere* Leerzeichen werden wie *ein* Leerzeichen behandelt. Wenn man andere als die normalen Wort- und Zeilenabstände will, kann man dies also nicht durch die Eingabe von zusätzlichen Leerzeichen oder Leerzeilen erreichen, sondern nur mit entprechenden LaTeX-Befehlen.

Eine Leerzeile zwischen Textzeilen bedeutet das Ende eines Absatzes. *Mehrere* Leerzeilen werden wie *eine* Leerzeile behandelt.

2.2 LaTeX-Befehle und Gruppen

Die meisten LaTeX-Befehle haben eines der beiden folgenden Formate: Entweder sie beginnen mit einem Backslash (\) und haben dann einen nur aus Buchstaben bestehenden Namen, der durch ein oder mehrere Leerzeichen oder durch ein nachfolgendes Sonderzeichen oder eine Ziffer beendet wird; oder sie bestehen aus einem Backslash und genau einem Sonderzeichen oder einer Ziffer. Groß- und Kleinbuchstaben haben auch in Befehlsnamen *verschiedene* Bedeutung. Wenn man nach einem Befehlsnamen eine Leerstelle erhalten will, muß man `{}` zur Beendigung des Befehlsnamens oder einen eigenen Befehl für die Leerstelle verwenden.

Heute ist der 35. Mai 1998. Oder: Heute ist der 35. Mai 1998. Falsch ist: Am 35. Mai 1998regnet es. Richtig: Am 35. Mai 1998 scheint die Sonne. Oder: Am 35. Mai 1998 schneit es.

```
Heute ist der \today.
Oder: Heute ist der \today .
Falsch ist:
 Am \today regnet es.
Richtig:
 Am \today{} scheint die Sonne.
 Oder: Am \today\ schneit es.
```

Manche Befehle haben Parameter, die zwischen geschwungenen Klammern angegeben werden müssen. Manche Befehle haben Parameter, die weggelassen oder zwischen eckigen Klammern angegeben werden können. Manche Befehle haben Varianten, die durch das Hinzufügen eines Sterns an den Befehlsnamen unterschieden werden.

Geschwungene Klammern können auch dazu verwendet werden, Gruppen (groups) zu bilden. Die Wirkung von Befehlen, die innerhalb von Gruppen oder Umgebungen (environments) angegeben werden, endet immer mit dem Ende der Gruppe bzw. der Umgebung. Im obigen Beispiel ist `{}` eine leere Gruppe, die außer der Beendigung des Befehlsnamens `today` keine Wirkung hat.

2.3 Kommentare

Alles, was hinter einem Prozentzeichen (`%`) steht (bis zum Ende der Eingabezeile), wird von LaTeX ignoriert. Dies kann für Notizen des Autors verwendet werden, die nicht oder noch nicht ausgedruckt werden sollen.

Das ist ein Beispiel.

```
Das ist ein % dummes
% Besser: ein lehrreiches <----
Beispiel.
```

2.4 Aufbau

Der erste Befehl in einem LaTeX-Eingabefile muß der Befehl

```
\documentclass
```

sein. Er legt fest, welche Art von Schriftstück überhaupt erzeugt werden soll (Bericht, Buch, Brief usw.). Danach können weitere Befehle folgen, die für das gesamte Dokument gelten sollen. Dieser Teil des Dokuments wird auch als *Vorspann* oder *Präambel* bezeichnet. Mit dem Befehl

```
\begin{document}
```

endet der Vorspann, und es beginnt das Setzen des Schriftstücks. Nun folgen der Text und alle LaTeX-Befehle, die das Ausdrucken des Schriftstücks bewirken. Die Eingabe muß mit dem Befehl

```
\end{document}
```

beendet werden. Falls nach diesem Befehl noch Eingaben folgen, werden sie von LaTeX ignoriert.

Abbildung 1 zeigt ein *minimales* LaTeX-File. Ein etwas komplizierteres File ist in Abbildung 2 skizziert.

```
\documentclass{article}
\begin{document}
Small is beautiful.
\end{document}
```

Abbildung 1: Ein minimales LaTeX-File

2.5 Dokumentklassen

Die am Beginn des Eingabefiles mit

\documentclass[*optionen*]{*klasse*}

definierte „Klasse“ eines Dokumentes enthält Vereinbarungen über das Layout und die logischen Strukturen, z. B. die Gliederungseinheiten (Kapitel etc.), die für alle Dokumente dieses Typs gemeinsam sind.

Zwischen den geschwungenen Klammern *muß* genau eine Dokumentklasse angegeben werden. Tabelle 1 auf S. 9 führt Klassen auf, die in jeder LaTeX-Installation existieren sollten. Im *Local Guide* [7] können weitere verfügbare Klassen angegeben sein.

```
\documentclass[11pt,a4paper]{article}
\usepackage[latin1]{inputenc}
\usepackage{ngerman}
\date{29. Februar 1998}
\author{H.~Partl}
\title{Über kurz oder lang}

\begin{document}
\maketitle
\begin{abstract}
Beispiel für einen wissenschaftlichen Artikel
in deutscher Sprache.
\end{abstract}
\tableofcontents

\section{Start}

Hier beginnt mein schönes Werk ...

\section{Ende}

... und hier endet es.

\end{document}
```

Abbildung 2: Aufbau eines Artikels

Zwischen den eckigen Klammern *können*, durch Kommata getrennt, eine oder mehrere Optionen für das Klassenlayout angegeben werden. Die wichtigsten Optionen sind in der Tabelle 2 auf S. 10 angeführt. Das Eingabefile für diese Beschreibung beginnt z. B. mit:

```
\documentclass[11pt,a4paper]{article}
```

2.6 Pakete

Mit dem Befehl

`\usepackage[`*optionen*`]{`*pakete*`}`

können im Vorspann ergänzende Makropakete (packages) geladen werden, die das Layout der Dokumentklasse modifizieren oder zusätzliche Funktionalität bereitstellen. Eine Auswahl von Paketen findet sich in der Tabelle 3 auf S. 11. Das Eingabefile für diese Beschreibung enthält beispielsweise:

```
\usepackage{german,latexsym,textcomp,alltt}
\usepackage[dvips,draft]{graphics}
```

Tabelle 1: Dokumentklassen

`article` für Artikel in wissenschaftlichen Zeitschriften, kürzere Berichte u. v. a.

`report` für längere Berichte, die aus mehreren Kapiteln bestehen, Diplomarbeiten, Dissertationen u. ä.

`book` für Bücher

`scrartcl, scrreprt, scrbook` sind Varianten der o. g. Klassen mit besserer Anpassung an DIN-Papierformate und „europäische“ Typographie. (Nicht überall vorhanden, siehe *Local Guide* [7].)

`proc` für Konferenzbände (Proceedings)

`letter` für Briefe (siehe auch Abschnitt 8.2)

`slides` für Folien. (Ersetzt das SLITEX-Format von LATEX 2.09.)

2.7 Eingabezeichensatz

Bei jedem LATEX-System dürfen mindestens die folgenden Zeichen zur Eingabe von Text verwendet werden:

```
a...z A...Z 0...9
.  :  ; , ?  !  ‘ ’ ( ) [ ] - / * @ + =
```

Die folgenden Eingabezeichen haben für LATEX eine Spezialbedeutung oder sind nur innerhalb von mathematischen Formeln erlaubt:

```
$ & % # _ { }  ~  ^  "  \  | < >
```

Für Zeichen, die über obige Liste hinausgehen, beispielsweise die Umlaute, sind je nach Betriebssystem des verwendeten Computers unterschiedliche Kodierungen in Gebrauch. Damit auch diese Zeichen im Eingabefile benutzt werden dürfen, muß man das Paket `inputenc` laden und dabei die jeweilige Kodierung als Option angeben: `\usepackage[`*codepage*`]{inputenc}`. Mögliche Angaben für *codepage* sind u. a.:

`latin1` ISO Latin-1, gebräuchlich auf UNIX-Systemen und VMS

`cp850` IBM Codepage 850, üblich unter OS/2

`ansinew` Latin-1 mit Erweiterungen à la WINDOWS

`cp437de` IBM Codepage 437, üblich unter DOS

`applemac` MACINTOSH-Kodierung

Tabelle 2: Klassenoptionen (Alternativen sind durch `|` getrennt)

`10pt|11pt|12pt` wählt die normale Schriftgröße des Dokuments aus. 10 pt hohe Schrift ist die Voreinstellung; diese Beschreibung benutzt 11 pt.

`a4paper` für Papier im DIN A4-Format. Ohne diese Option nimmt LaTeX amerikanisches Papierformat an.

`fleqn` für linksbündige statt zentrierte mathematische Gleichungen

`leqno` für Gleichungsnummern links statt rechts von jeder numerierten Gleichung

`titlepage|notitlepage` legt fest, ob Titel und Zusammenfassung auf einer eigenen Seite erscheinen sollen. `titlepage` ist die Voreinstellung für die Klassen `report` und `book`.

`onecolumn|twocolumn` für ein- oder zweispaltigen Satz. Die Voreinstellung ist immer `onecolumn`. Die Klassen `letter` und `slides` kennen *keinen* zweispaltigen Satz.

`oneside|twoside` legt fest, ob die Seiten für ein- oder zweiseitigen Druck gestaltet werden sollen. `oneside` ist die Voreinstellung für alle Klassen außer `book`.

(Es ist möglich, daß bestimmte LaTeX-Implementierungen diese Fähigkeit nicht unterstützen; ziehen Sie dazu den *Local Guide* [7] zu Rate.) Falls LaTeX ein eingegebenes Zeichen nicht darstellen kann, was meist für die sogenannten „Pseudografik-Zeichen“ gilt, bekommt man eine entsprechende Fehlermeldung. Auch sind manche Zeichen nur im Text, andere nur in mathematischen Formeln erlaubt.

Man beachte, daß der in der *Ausgabe* darstellbare Zeichenvorrat von LaTeX nicht davon abhängt, welche Zeichen als *Eingabe* erlaubt sind: Für jedes überhaupt darstellbare Zeichen – also auch diejenigen, die nicht im Zeichensatz des jeweiligen Betriebssystems enthalten sind – gibt es einen LaTeX-Befehl oder eine Ersatzdarstellung, die ausschließlich mit ASCII-Zeichen auskommt. Näheres darüber erfahren Sie in Abschnitt 3.4.

Tabelle 3: Pakete (eine Auswahl)

`alltt` Definiert eine Variante der `verbatim`-Umgebung

`amsmath, amssymb` Mathematischer Formelsatz mit erweiterten Fähigkeiten, zusätzliche mathematische Schriften und Symbole; Beschreibung siehe [5].

`array` Verbesserte und erweiterte Versionen der Umgebungen `array`, `tabular` und `tabular*`.

`babel` Anpassungen für viele verschiedene Sprachen. Die gewählten Sprachen werden als Optionen angegeben.

`color` Unterstützung für Farbdruck (nur mit bestimmten Druckertreibern); Beschreibung siehe [11] und [6].

`dcolumn` Unterstützt auf Dezimaltrennzeichen ausgerichtete Spalten in den Umgebungen `array` und `tabular`

`fontenc` Erlaubt die Verwendung von Schriften mit unterschiedlicher Kodierung (Zeichenvorrat, Anordnung).

`fancyhdr` Flexible Gestaltung von Kopf- und Fußzeilen.

`german, ngerman` Anpassungen für die deutsche Sprache in traditioneller und neuer Rechtschreibung.

`graphics` Einbindung von extern erzeugten Graphiken. Die umfangreichen Möglichkeiten dieses Pakets werden in [11] und [6] beschrieben.

`inputenc` Deklaration der Zeichenkodierung im Eingabefile.

`latexsym` Erlaubt einige besondere Symbole wie $\Box$, die mit LaTeX 2.09 standardmäßig verfügbar waren.

`longtable` für Tabellen über mehrere Seiten mit automatischem Seitenumbruch.

`makeidx` Unterstützt das Erstellen eines Index.

`multicol` Mehrspaltiger Satz mit Kolumnenausgleich.

`showkeys` Druckt die Namen aller verwendeten `\label`s, `\ref`s und `\pageref`s im Text aus.

`textcomp` Bindet Schriften mit zusätzlichen Textsymbolen ein.

3 Setzen von Text

3.1 Deutschsprachige Texte

LaTeX wurde ursprünglich für den englischen Sprachraum entwickelt. Für Texte, die in einer anderen Sprache als (amerikanischem) Englisch verfaßt sind, muß deshalb ein zusätzliches Paket (siehe Abschnitt 2.5) zur Sprachanpassung geladen werden. Für deutschsprachige Texte ist das normalerweise das Paket `german` oder `ngerman`:

```
\usepackage{german}
```

oder

```
\usepackage{ngerman}
```

Für Texte in *traditioneller* Rechtschreibung ist `german` zu benutzen, für Texte in der *neuen* deutschen Rechtschreibung `ngerman`. Der Grund für diese Unterscheidung ist die unterschiedliche Silbentrennung. Eine ausführliche Beschreibung dieser Pakete findet man in [13]. Wenn im folgenden vom Paket `german` die Rede ist, so bezieht sich das normalerweise auch auf `ngerman`.

3.2 Zeilen- und Seiten-Umbruch

3.2.1 Blocksatz

Normaler Text wird im Blocksatz, d. h. mit Randausgleich gesetzt. LaTeX führt den Zeilen- und Seitenumbruch automatisch durch. Dabei wird für jeden Absatz die bestmögliche Aufteilung der Wörter auf die Zeilen bestimmt, und wenn notwendig werden Wörter automatisch abgeteilt.

Das Ende von Wörtern und Sätzen wird durch Leerzeichen gekennzeichnet. Hierbei spielt es keine Rolle, ob man ein oder 100 Leerzeichen eingibt.

Eine oder mehrere Leerzeilen kennzeichnen das Ende von Absätzen.

```
Das Ende von Wörtern und
Sätzen wird durch Leerzeichen
gekennzeichnet.
Hierbei spielt es keine Rolle,
ob man ein  oder          100
Leerzeichen eingibt.

Eine oder mehrere Leerzeilen
kennzeichnen das Ende von
Absätzen.
```

Wie die Absätze gesetzt werden, hängt von der Dokumentklasse ab: Die Klassen `article`, `report` und `book` kennzeichnen Absätze durch Einrücken der ersten Zeile; die Klassen `letter` und `slides` lassen stattdessen zwischen den Absätzen einen kleinen vertikalen Abstand.

Mit Hilfe der in Abschnitt 3.9 beschriebenen Umgebungen ist es möglich, spezielle Textteile jeweils anders zu setzen.

Für Ausnahmefälle kann man den Umbruch außerdem mit den folgenden Befehlen beeinflussen: Der Befehl `\\` oder `\newline` bewirkt einen Zeilenwechsel ohne neuen Absatz, der Befehl `\\*` einen Zeilenwechsel, bei dem kein Seitenwechsel erfolgen darf. Der Befehl `\newpage` bewirkt einen Seitenwechsel.

Mit den Befehlen `\linebreak[`*n*`]`, `\nolinebreak[`*n*`]`, `\pagebreak[`*n*`]` und `\nopagebreak[`*n*`]` kann man angeben, ob an bestimmten Stellen ein Zeilen- bzw. Seitenwechsel eher günstig oder eher ungünstig ist, wobei n die Stärke der Beeinflussung angibt (1, 2, 3 oder 4).

Mit dem LaTeX-Befehl `\enlargethispage{`*Länge*`}` läßt sich eine gegebene Seite um einen festen Betrag verlängern oder verkürzen. Damit ist es möglich, noch eine Zeile mehr auf eine Seite zu bekommen.

LaTeX bemüht sich, den Zeilenumbruch besonders schön zu machen. Falls es keine den strengen Regeln genügende Möglichkeit für einen glatten rechten Rand findet, läßt es eine Zeile zu lang und gibt eine entsprechende Fehlermeldung aus (`overfull hbox`). Das tritt insbesondere dann auf, wenn keine geeignete Stelle für die Silbentrennung gefunden wird. Innerhalb der `sloppypar`-Umgebung ist LaTeX generell weniger streng in seinen Ansprüchen und vermeidet solche überlange Zeilen, indem es die Wortabstände stärker – notfalls auch unschön – vergrößert. In diesem Fall werden zwar Warnungen gemeldet (`underfull hbox`), das Ergebnis ist aber meistens durchaus brauchbar.

3.2.2 Silbentrennung

Falls die automatische Silbentrennung in einzelnen Fällen nicht das richtige Ergebnis liefert, kann man diese Ausnahmen mit den folgenden Befehlen richtigstellen. Das kann insbesondere bei zusammengesetzten oder fremdsprachigen Wörtern notwendig werden; außerdem findet LaTeX in Wörtern mit Umlauten oder akzentuierten Buchstaben nicht alle zulässigen Trennstellen.

Der Befehl `\hyphenation` bewirkt, daß die darin angeführten Wörter jedesmal an den und nur an den mit `-` markierten Stellen abgeteilt werden können. Er sollte im Vorspann stehen und eignet sich *nur* für Wörter, die keine Umlaute, scharfes s, Ziffern oder sonstige Sonderzeichen enthalten.

```
\hyphenation{ Eingabe-file
   Eingabe-files FORTRAN }
```

Der Befehl `\-` innerhalb eines Wortes bewirkt, daß dieses Wort dieses eine Mal nur an den mit `\-` markierten Stellen oder unmittelbar nach einem Bindestrich abgeteilt werden kann. Dieser Befehl eignet sich für *alle* Wörter, auch für solche, die Umlaute, scharfes s, Ziffern oder sonstige Sonderzeichen enthalten.

Eingabefile, LaTeX-Eingabe- file, Häßlichkeit	`Ein\-gabe\-file,` `\LaTeX-Eingabe\-file,` `Häß\-lich\-keit`

Der Befehl `\mbox{...}` bewirkt, daß das Argument überhaupt nicht abgeteilt werden kann.

Die Telefonnummer ist nicht mehr (02 22) 56 01-36 94. *filename* gibt den Dateinamen an.	`Die Telefonnummer ist nicht mehr` `\mbox{(02\,22) 56\,01-36\,94}. \\` `\mbox{\textit{filename}} gibt den` `Dateinamen an.`

Innerhalb des von `\mbox` eingeschlossenen Text können Wortabstände für den notwendigen Randausgleich bei Blocksatz nicht mehr verändert werden. Ist dies nicht erwünscht, sollte man besser einzelne Wörter oder Wortteile in `\mbox` einschließen und diese mit einer Tilde `~`, einem untrennbaren Wortzwischenraum (siehe Abschnitt 3.3), verbinden.

Das Paket `german` macht noch einige weitere Befehle verfügbar, die bestimmte Besonderheiten der deutschen Sprache berücksichtigen. Die wichtigsten von ihnen sind: `"ck` für „ck“, das als „k-k“ abgeteilt wird, `"ff` für „ff“, das als „ff-f“ abgeteilt wird (und ebenso für andere Konsonanten) und `"~` für einen Bindestrich, an dem kein Zeilenumbruch stattfinden soll.

Drucker bzw. Druk-ker	`Dru"cker`
Rolladen bzw. Roll-laden	`Ro"lladen`
x-beliebig	`x"~beliebig`
bergauf und -ab	`bergauf und "~ab`

3.3 Wortabstand

Um einen glatten rechten Rand zu erreichen, variiert LaTeX die Leerstellen zwischen den Wörtern etwas. Nach Punkten, Fragezeichen u. a., die einen Satz beenden, wird dabei ein etwas größerer Abstand erzeugt, was die Lesbarkeit des Textes erhöht. LaTeX nimmt an, daß Punkte, die auf einen Großbuchstaben folgen, eine Abkürzung bedeuten, und daß alle anderen Punkte einen Satz beenden. Ausnahmen von diesen Regeln muß man LaTeX mit den folgenden Befehlen mitteilen:

Ein Backslash (\) vor einem Leerzeichen bedeutet, daß diese Leerstelle nicht verbreitert werden darf.

Eine `~` (Tilde) bedeutet eine Leerstelle, an der kein Zeilenwechsel erfolgen darf.

Mit `\,` läßt sich ein kurzer Abstand erzeugen, wie er z. B. in Abkürzungen vorkommt oder zwischen Zahlenwert und Maßeinheit.

Der Befehl `\@` vor einem Punkt bedeutet, daß dieser Punkt einen Satz beendet, obwohl davor ein Großbuchstabe steht.

Das betrifft u. a. auch die
wissenschaftl. Mitarbeiter.
Dr. Partl wohnt im 1. Stock.
… 5 cm breit.
Ich brauche Vitamin C. Du nicht?

```
Das betrifft u.\,a.\ auch die \\
wissenschaftl.\ Mitarbeiter. \\
Dr.~Partl wohnt im 1.~Stock. \\
\dots\ 5\,cm breit. \\
Ich brauche Vitamin~C\@.
Du nicht?
```

Außerdem gibt es die Möglichkeit, mit dem Befehl `\frenchspacing` zu vereinbaren, daß die Abstände an Satzenden nicht anders behandelt werden sollen als die zwischen Wörtern. Diese Konvention ist im nicht-englischen Sprachraum verbreitet. In diesem Fall brauchen die Befehle `\ ` und `\@` nicht angegeben werden. Mit dem Paket `german` ist `\frenchspacing` automatisch gewählt; das kann durch `\nonfrenchspacing` wieder rückgängig gemacht werden – so wie durchgängig im vorliegenden Dokument!

3.4 Spezielle Zeichen

3.4.1 Anführungszeichen

Für Anführungszeichen ist *nicht* das auf Schreibmaschinen übliche Zeichen (`"`) zu verwenden. Im Buchdruck werden für öffnende und schließende Anführungszeichen jeweils verschiedene Zeichen bzw. Zeichenkombinationen gesetzt. Öffnende Anführungszeichen, wie sie im amerikanischen Englisch üblich sind, erhält man durch Eingabe von zwei Grave-Akzenten, schließende durch zwei Apostrophe.

“No,” he said, “I don’t know!”	`‘‘No,’’ he said,` `‘‘I don’t know!’’`

„Deutsche Gänsefüßchen“ sehen anders aus als “amerikanische Quotes”. Bei Benutzung des Paketes `german` (siehe 3.1) stehen die folgenden Befehle für deutsche Anführungszeichen zur Verfügung: `"‘` (Doublequote und Grave-Akzent) für Anführungszeichen unten, und `"’` (Doublequote und Apostroph) für Anführungszeichen oben.

„Nein,“ sagte er, „ich weiß nichts!“	`"‘Nein,"’ sagte er,` `"‘ich weiß nichts!"’`

In den Zeichensätzen mancher Rechner (z. B. Macintosh) sind die deutschen Anführungszeichen enthalten. Das Paket `inputenc` (siehe Abschnitt 2.7) erlaubt dann, sie auch direkt einzugeben.

3.4.2 Binde- und Gedankenstriche

Im Buchdruck werden verschiedene Striche für Bindestriche, Gedankenstriche und Minus-Zeichen verwendet. Die verschieden langen Striche werden in LaTeX durch Kombinationen von Minus-Zeichen angegeben. Der ganz lange Gedankenstrich (—) wird im Deutschen nicht benutzt, im Englischen wird er ohne Leerzeichen eingefügt.

O-Beine	`O-Beine`
10–18 Uhr	`10--18~Uhr`
Paris–Dakar	`Paris--Dakar`
Schalke 04 – Hertha BSC	`Schalke 04 -- Hertha BSC`
ja – oder nein?	`ja -- oder nein?`
yes—or no?	`yes---or no?`
0, 1 und -1	`0, 1 und $-1$`

3.4.3 Punkte

Im Gegensatz zur Schreibmaschine, wo jeder Punkt und jedes Komma mit einem der Buchstabenbreite entsprechenden Abstand versehen ist, werden Punkte und Kommata im Buchdruck eng an das vorangehende Zeichen gesetzt. Für Fortsetzungspunkte (drei Punkte mit geeignetem Abstand) gibt es daher einen eigenen Befehl `\ldots` oder `\dots`.

Nicht so ... sondern so:	`Nicht so ... sondern so: \\`
Wien, Graz, …	`Wien, Graz, \dots`

3.4.4 Ligaturen

Im Buchdruck ist es üblich, manche Buchstabenkombinationen anders zu setzen als die Einzelbuchstaben.

ﬀ ﬁ ﬂ AV Te … statt ff fi fl AV Te …

Mit Rücksicht auf die Lesbarkeit des Textes sollten diese Ligaturen und Unterschneidungen (kerning) unterdrückt werden, wenn die betreffenden Buchstabenkombinationen nach Vorsilben oder bei zusammengesetzten Wörtern zwischen den Wortteilen auftreten. Dazu dient der Befehl `\/`.

Nicht Auﬂage (Au-ﬂ-age)	`Nicht Auflage (Au-fl-age) \\`
sondern Auflage (Auf-lage)	`sondern Auf\/lage (Auf-lage)`

Mit dem Paket `german` steht zusätzlich der Befehl `"|` zur Verfügung, der gleichzeitig eine Trennhilfe darstellt.

Auflage (Auf-lage)	`Auf"|lage (Auf-lage)`

3.4.5 Symbole, Akzente und besondere Buchstaben

Einige der Zeichen, die bei der Eingabe eine Spezialbedeutung haben, können durch das Voranstellen des Zeichens `\` (Backslash) ausgedruckt werden:

$ & % # _ { }	`\$ \& \% \# \_ \{ \}`

Für andere gibt es besondere Befehle. Sie gelten nur für normalen Text; wie derartige Symbole innerhalb von mathematischen Formeln gesetzt werden, erfahren Sie im Kapitel 4:

˜	`\textasciitilde`
ˆ	`\textasciicircum`
\	`\textbackslash`
|	`\textbar`
$<$	`\textless`
$>$	`\textgreater`

LaTeX ermöglicht darüber hinaus die Verwendung von Akzenten und speziellen Buchstaben aus zahlreichen verschiedenen Sprachen, siehe die Tabellen 4 und 5. Akzente werden darin jeweils am Beispiel des Buchstabens o gezeigt, können aber prinzipiell auf jeden Buchstaben gesetzt werden. Wenn ein Akzent auf ein i oder j gesetzt werden soll, muß der i-Punkt wegbleiben. Dies erreicht man mit den Befehlen `\i` und `\j`. Es steht auch ein Befehl `\textcircled` für eingekreiste Zeichen zur Verfügung.

Tabelle 4: Akzente und spezielle Buchstaben

ò	`\`o`	ó	`\'o`	ô	`\^o`
õ	`\~o`	ō	`\=o`	ȯ	`\.o`
ŏ	`\u o`	ǒ	`\v o`	ő	`\H o`
ö	`\"o`	o̧	`\c o`	ọ	`\d o`
o̲	`\b o`	o̊	`\r o`	o͡o	`\t oo`
œ	`\oe`	Œ	`\OE`	æ	`\ae`
Æ	`\AE`	å	`\aa`	Å	`\AA`
ø	`\o`	Ø	`\O`	ł	`\l`
Ł	`\L`	ı	`\i`	ȷ	`\j`
ß	`\ss`				

Tabelle 5: Symbole

¡	`!``	†	`\dag`	™	`\texttrademark`
¿	`?``	‡	`\ddag`	·	`\textperiodcentered`
§	`\S`	¶	`\P`	•	`\textbullet`
£	`\pounds`	©	`\copyright`	®	`\textregistered`

Hôtel, naïve, smørebrød.	`H\^otel, na\"\i ve,` `sm\o rebr\o d.`
Die häßliche Straße.	`Die h\"a\ss{}liche` `Stra\ss{}e.`
¡Señorita!	`!`Se\~norita!`
ⓧ	`\textcircled{x}`

Benutzt man das Paket `inputenc` mit der passenden Option für das jeweilige Betriebssystem, siehe Abschnitt 2.7, dann darf man diese Zeichen – soweit sie im Zeichensatz des Betriebssystems existieren – auch direkt in das Eingabefile schreiben.

Mit dem Paket `german` können Umlaute auch durch einfaches Voranstellen eines Doublequotes geschrieben werden, also z. B. `"o` für „ö“; für scharfes s darf man `"s` schreiben (ohne Probleme mit nachfolgenden Leerstellen):

Die häßliche Straße muß schöner werden.	`Die h"a"sliche Stra"se` `mu"s sch"oner werden.`

Diese Notation wurde eingeführt, als die direkte Eingabe und Anzeige von Umlauten auf vielen Rechnersystemen noch nicht möglich war. Als Quasi-Standard zum plattformübergreifenden Austausch von TeX- und LaTeX-Dokumenten ist sie aber nach wie vor nützlich und für deutschsprachige Texte weit verbreitet; sie wird auch in einigen Beispielen dieser Kurzbeschreibung benutzt.

3.5 Kapitel und Überschriften

Der Beginn eines Kapitels bzw. Unterkapitels und seine Überschrift werden mit Befehlen der Form `\section{...}` angegeben. Dabei muß die logische Hierarchie eingehalten werden.
Bei der Klasse `article`:

```
\section  \subsection  \subsubsection
```

Bei den Klassen `report` und `book`:

```
\chapter  \section  \subsection  \subsubsection
```

Artikel können also relativ einfach als Kapitel in ein Buch eingebaut werden. Die Abstände zwischen den Kapiteln, die Numerierung und die Schriftgröße der Überschrift werden von LaTeX automatisch bestimmt.

Die Überschrift des gesamten Artikels bzw. die Titelseite des Schriftstücks wird mit dem Befehl `\maketitle` gesetzt. Der Inhalt muß vorher mit den Befehlen `\title`, `\author` und `\date` vereinbart werden (vgl. Abbildung 2 auf Seite 8).

Der Befehl `\tableofcontents` bewirkt, daß ein Inhaltsverzeichnis ausgedruckt wird. LaTeX nimmt dafür immer die Überschriften und Seitennummern von der jeweils letzten vorherigen Verarbeitung des Eingabefiles. Bei einem neu erstellten oder um neue Kapitel erweiterten Schriftstück muß man das Programms LaTeX also mindestens zweimal aufrufen, damit man die richtigen Angaben erhält.

Es gibt auch Befehle der Form `\section*{...}`, bei denen keine Numerierung und keine Eintragung ins Inhaltsverzeichnis erfolgen.

Mit den Befehlen `\label` und `\ref` ist es möglich, die von LaTeX automatisch vergebenen Kapitelnummern im Text anzusprechen. Für `\ref{...}` setzt LaTeX die mit `\label{...}` definierte Nummer ein. Auch hier wird immer die Nummer von der letzten vorherigen Verarbeitung des Eingabefiles genommen. Beispiel:

```
\section{Algorithmen}
...
Der Beweis findet sich in Abschnitt~\ref{bew}.
...
\section{Beweise} \label{bew}
...
```

3.6 Fußnoten

Fußnoten[1] werden automatisch numeriert und am unteren Ende der Seite ausgedruckt. Innnerhalb von Gleitobjekten (siehe Abschnitt 6.2), Tabellen (3.9.6) oder der `tabbing`-Umgebung (3.9.5) ist der Befehl `\footnote` nicht erlaubt.

```
Fußnoten\footnote{Das ist eine
Fußnote.} werden ...
```

[1] Das ist eine Fußnote.

3.7 Hervorgehobener Text

In maschinengeschriebenen Texten werden hervorzuhebende Texte unterstrichen, im Buchdruck werden dafür verschiedene Schriftarten verwendet. Der Befehl

`\emph{`*text*`}`

(emphasize) setzt seinen Parameter in einer auffälligen Schriftart. LaTeX verwendet für den hervorgehobenen Text *kursive* Schrift.

Werden innerhalb eines hervorgehobenen Textes nochmals *Passagen hervorgehoben, so setzt LaTeX diese in einer* aufrechten *Schrift.*

```
\emph{Werden innerhalb eines
hervorgehobenen Textes
\emph{nochmals} Passagen
hervorgehoben, so setzt
\LaTeX\ diese in einer
\emph{aufrechten} Schrift.}
```

3.8 Hochgestellter Text

Hochgestellten Text in passender Größe generiert folgender Befehl:

`\textsuperscript{`*text*`}`

le 2$^{\text{ième}}$ régime

```
le 2\textsuperscript{i\`eme}
r\'egime
```

3.9 Umgebungen

Die Kennzeichnung von speziellen Textteilen, die anders als im normalen Blocksatz gesetzt werden sollen, erfolgt mittels sogenannter Umgebungen (environments) in der Form

`\begin{`*name*`}` *text* `\end{`*name*`}`

Umgebungen sind *Gruppen*. Sie können auch ineinander geschachtelt werden, dabei muß aber die richtige Reihenfolge beachtet werden:

```
\begin{aaa}...\begin{bbb}...\end{bbb}...\end{aaa}
```

3.9.1 Zitate (quote, quotation, verse)

Die `quote`-Umgebung eignet sich für kürzere Zitate, hervorgehobene Sätze und Beispiele. Der Text wird links und rechts eingerückt.

Eine typographische Faustregel für die Zeilenlänge lautet:

> Keine Zeile soll mehr als ca. 66 Buchstaben enthalten.

Deswegen werden in Zeitungen mehrere Spalten nebeneinander verwendet.

```
Eine typographische Faustregel
für die Zeilenlänge lautet:
\begin{quote}
Keine Zeile soll mehr als
ca.\ 66~Buchstaben enthalten.
\end{quote}
Deswegen werden in Zeitungen
mehrere Spalten nebeneinander
verwendet.
```

Die `quotation`-Umgebung unterscheidet sich in den Standardklassen (vgl. Tabelle 1 auf Seite 9) von der `quote`-Umgebung dadurch, daß Absätze durch Einzüge gekennzeichnet werden. Sie ist daher für längere Zitate, die aus mehreren Absätzen bestehen, geeignet.

Die `verse`-Umgebung eignet sich für Gedichte und für Beispiele, bei denen die Zeilenaufteilung wesentlich ist. Die Verse (Zeilen) werden durch `\\` getrennt, Strophen durch Leerzeilen.

3.9.2 Listen (itemize, enumerate, description)

Die Umgebung `itemize` eignet sich für einfache Listen (siehe Abbildung 3). Die Umgebung `enumerate` eignet sich für numerierte Aufzählungen (siehe Abbildung 4). Die Umgebung `description` eignet sich für Beschreibungen (siehe Abbildung 5).

Listen:

- Bei `itemize` werden die Elemente durch Punkte und andere Symbole gekennzeichnet.
- Listen kann man auch verschachteln:
  - Die maximale Schachtelungstiefe ist 4.
  - Bezeichnung und Einrückung der Elemente wechseln automatisch.
- usw.

```
Listen:
\begin{itemize}

\item Bei \texttt{itemize}
werden die Elemente ...

\item Listen kann man auch
verschachteln:
  \begin{itemize}
  \item Die maximale ...
  \item Bezeichnung und ...
  \end{itemize}

\item usw.

\end{itemize}
```

Abbildung 3: Beispiel für `itemize`

Numerierte Listen:

1. Bei `enumerate` werden die Elemente mit Ziffern oder Buchstaben numeriert.
2. Die Numerierung erfolgt automatisch.
3. Listen kann man auch verschachteln:
    (a) Die maximale Schachtelungstiefe ist 4.
    (b) Bezeichnung und Einrükkung der Elemente wechseln automatisch.
4. usw.

```
Numerierte Listen:
\begin{enumerate}

\item Bei \texttt{enumerate}
werden die Elemente ...

\item Die Numerierung ...

\item Listen kann man auch
verschachteln:
  \begin{enumerate}
  \item Die maximale ...
  \item Bezeichnung und ...
  \end{enumerate}

\item usw.

\end{enumerate}
```

Abbildung 4: Beispiel für `enumerate`

Kleine Tierkunde:

Gelse: ein kleines Tier, das östlich des Semmering Touristen verjagt.

Gemse: ein großes Tier, das westlich des Semmering von Touristen verjagt wird.

Gürteltier: ein mittelgroßes Tier, das hier nur wegen der Länge seines Namens vorkommt.

```
Kleine Tierkunde:
\begin{description}
\item[Gelse:]
   ein kleines Tier, das ...
\item[Gemse:]
   ein großes Tier, das ...
\item[Gürteltier:]
   ein mittelgroßes Tier,
   das ...
\end{description}
```

Abbildung 5: Beispiel für `description`

3.9.3 Linksbündig, rechtsbündig, zentriert (flushleft, flushright, center)

Die Umgebungen `flushleft` und `flushright` bewirken links- bzw. rechtsbündigen Satz ohne Randausgleich („Flattersatz") und ohne Trennungen, `center` setzt den Text in die Mitte der Zeile. Die einzelnen Zeilen werden durch `\\` getrennt. Wenn man `\\` nicht angibt, bestimmt LaTeX automatisch die Zeilenaufteilung.

links Backbord	`\begin{flushleft}` `links \\` `Backbord` `\end{flushleft}`
rechts Steuerbord	`\begin{flushright}` `rechts  \\` `Steuerbord` `\end{flushright}`
Im Reich der Mitte	`\begin{center}` `Im \\ Reich \\ der \\ Mitte` `\end{center}`

3.9.4 Direkte Ausgabe (verbatim, verb)

Zwischen `\begin{verbatim}` und `\end{verbatim}` stehende Zeilen werden genauso ausgedruckt, wie sie eingegeben wurden, d. h. mit allen Leerzeichen und Zeilenwechseln und ohne Interpretation von Spezialzeichen und LaTeX-Befehlen. Dies eignet sich z. B. für das Ausdrucken eines (kurzen) Computer-Programms.

Innerhalb eines Absatzes können einzelne Zeichenkombinationen oder kurze Textstücke ebenso „wörtlich" ausgedruckt werden, indem man sie zwischen `\verb|` und `|` einschließt. Mit diesen Befehlen wurden z. B. alle LaTeX-Befehle in der vorliegenden Beschreibung gesetzt.

Der `\dots`-Befehl …	`Der \verb|\dots|-Befehl \dots`

Die `verbatim`-Umgebung und der Befehl `\verb` dürfen *nicht* innerhalb von Parametern von anderen Befehlen verwendet werden.

3.9.5 Tabulatoren (tabbing)

In der `tabbing`-Umgebung kann man Tabulatoren ähnlich wie an Schreibmaschinen setzen und verwenden. Der Befehl `\=` setzt eine Tabulatorposition, `\kill` bedeutet, daß die „Musterzeile" nicht ausgedruckt werden soll, `\>` springt zur nächsten Tabulatorposition, und `\\` trennt die Zeilen.

links	Mittelteil	rechts
Es		
war einmal	und ist	nicht mehr
ein		ausgestopfter
		Teddybär

```
\begin{tabbing}
war einmal\quad \=
 Mittelteil\quad \= \kill
links \> Mittelteil \> rechts\\
Es \\
war einmal \> und ist
 \> nicht mehr\\
ein  \>  \> ausgestopfter\\
 \>  \> Teddyb"ar
\end{tabbing}
```

3.9.6 Tabellen (tabular)

Die `tabular`-Umgebung dient zum Setzen von Tabellen, bei denen LaTeX automatisch die benötigte Spaltenbreite bestimmt, und bei der auch spezielle Eigenschaften wie Rechtsbündigkeit und Hilfslinien vereinbart werden können.

Im Parameter des Befehls `\begin{tabular}{...}` wird das Format der Tabelle angegeben. Dabei bedeutet `l` eine Spalte mit linksbündigem Text, `r` eine mit rechtsbündigem, `c` eine mit zentriertem Text, `p{`*breite*`}` eine Spalte der angegebenen Breite mit mehrzeiligem Text, `|` einen senkrechten Strich.

Innerhalb der Tabelle bedeutet `&` den Sprung in die nächste Tabellenspalte, `\\` trennt die Zeilen, `\hline` (an Stelle einer Zeile) setzt einen waagrechten Strich.

7C0	hexadezimal
3700	oktal
11111000000	binär
1984	dezimal

```
\begin{tabular}{|rl|}
\hline
7C0 & hexadezimal \\
3700 & oktal \\
11111000000 & bin"ar \\
\hline\hline
1984 & dezimal \\
\hline
\end{tabular}
```

4 Setzen von mathematischen Formeln

4.1 Allgemeines

Mathematische Textteile innerhalb eines Absatzes werden zwischen `\(` und `\)` oder zwischen `$` und `$` oder zwischen `\begin{math}` und `\end{math}` eingeschlossen. Als mathematische Texte gelten sowohl komplette mathematische Formeln als auch einzelne Variablennamen, die sich auf Formeln beziehen, griechische Buchstaben und diverse Sonderzeichen.

Seien a und b die Katheten und c die Hypotenuse, dann gilt $c^2 = a^2 + b^2$ (Satz des Pythagoras).

```
Seien \(a\) und \(b\) die Katheten
und \(c\) die Hypotenuse,
dann gilt \(c^{2}=a^{2}+b^{2}\)
(Satz des Pythagoras).
```

TeX spricht man wie $\tau\epsilon\chi$ aus.

Mit $\heartsuit$-lichen Grüßen

```
\TeX\ spricht man wie
 \(\tau\epsilon\chi\) aus.\\
Mit \(\heartsuit\)-lichen
 Gr"u"sen
```

Größere mathematische Formeln oder Gleichungen setzt man besser in eigene Zeilen. Wenn sie *keine* Gleichungsnummer erhalten sollen, stellt man sie dazu zwischen `\begin{displaymath}` und `\end{displaymath}` oder zwischen `\[` und `\]`; wenn sie eine Gleichungsnummer erhalten sollen, stellt man sie zwischen `\begin{equation}` und `\end{equation}`.

Seien a und b die Katheten und c die Hypotenuse, dann gilt

$$c = \sqrt{a^2 + b^2} \tag{1}$$

(Satz des Pythagoras).

```
Seien \(a\) und \(b\) die Katheten
und \(c\) die Hypotenuse,
dann gilt
\begin{equation}
c = \sqrt{  a^{2}+b^{2}  }
\end{equation}
(Satz des Pythagoras).
```

Mit `\label` und `\ref` kann man die Gleichungsnummern im Text ansprechen.

$$\varepsilon > 0 \tag{2}$$

Aus (2) folgt ...

```
\begin{equation} \label{eps}
\varepsilon > 0
\end{equation}

Aus (\ref{eps}) folgt \dots
```

Das Setzen im mathematischen Modus unterscheidet sich vom Text-Modus vor allem durch folgende Punkte:

1. Leerzeilen sind verboten (Mathematische Formeln müssen innerhalb eines Absatzes stehen).

2. Leerstellen und Zeilenwechsel haben bei der Eingabe keine Bedeutung, alle Abstände werden nach der Logik der mathematischen Ausdrücke automatisch bestimmt oder müssen durch spezielle Befehle wie `\,` oder `\qquad` angegeben werden.

$$\forall x \in \mathbf{R}: \qquad x^{2} \geq 0 \tag{3}$$

```
\begin{equation}
\forall x \in \mathbf{R}:
\qquad x^{2} \geq 0
\end{equation}
```

3. Jeder einzelne Buchstabe wird als Name einer Variablen betrachtet und entsprechend gesetzt (kursiv mit zusätzlichem Abstand). Will man innerhalb eines mathematischen Textes normalen Text (in aufrechter Schrift, mit Wortabständen) setzen, muß man diesen in `\textnormal{...}` einschließen.

$$x^{2} \geq 0 \qquad \textnormal{für alle } x \in \mathbf{R} \tag{4}$$

```
\begin{equation}
x^{2} \geq 0\qquad
\textnormal{f"ur alle }
x \in \mathbf{R}
\end{equation}
```

4.2 Elemente in mathematischen Formeln

In diesem Abschnitt werden die wichtigsten Elemente, die in mathematischen Formeln verwendet werden, kurz beschrieben. Eine Liste aller verfügbaren Symbole enthält Kapitel 4.5.

Kleine **griechische Buchstaben** werden als `\alpha`, `\beta`, `\gamma`, `\delta`, usw. eingegeben, große griechische Buchstaben als `\mathrm{A}`, `\mathrm{B}`, `\Gamma`, `\Delta`, usw.

$\lambda, \xi, \pi, \mu, \Phi, \Omega$

```
\(\lambda, \xi, \pi, \mu,
 \Phi, \Omega \)
```

Weiters gibt es eine Fülle von **mathematischen Symbolen**: von $\in$ über $\Rightarrow$ bis ∞ (siehe Kapitel 4.5).

Mathematische Symbole können in LaTeX 2ε aus einer Auswahl von **Alphabeten** genommen werden:

ABCabc	`\(\mathrm{ABCabc}\)`
$\mathbf{ABCabc}$	`\(\mathbf{ABCabc}\)`
ABCabc	`\(\mathsf{ABCabc}\)`
$\mathtt{ABCabc}$	`\(\mathtt{ABCabc}\)`
$\mathcal{ABC}$	`\(\mathcal{ABC}\)`

Die kalligraphischen Buchstaben (`\mathcal`) gibt es nur als Großbuchstaben. Lokal können weitere Alphabete, z. B. Fraktur, zur Verfügung stehen (siehe *Local Guide* [7]).

Exponenten und Indizes können mit den Zeichen `^` und `_` hoch- bzw. tiefgestellt werden.

$a_1 \qquad x^2 \qquad e^{-\alpha t} \qquad a^3_{ij}$

```
\(a_{1}\) \qquad
\(x^{2}\) \qquad
\(e^{-\alpha t}\) \qquad
\(a^{3}_{ij}\)
```

Das **Wurzelzeichen** wird mit `\sqrt`, n-te Wurzeln werden mit `\sqrt[`n`]` eingegeben. Die Größe des Wurzelzeichens wird von LaTeX automatisch gewählt.

$\sqrt{x} \qquad \sqrt{x^{2}+\sqrt{y}} \qquad \sqrt[3]{2}$

```
\(\sqrt{x}\) \qquad
\(\sqrt{ x^{2}+\sqrt{y} }\)
\qquad \(\sqrt[3]{2}\)
```

Die Befehle `\overline` und `\underline` bewirken **waagrechte Striche** direkt über bzw. unter einem Ausdruck.

$\overline{m+n}$

```
\(\overline{m+n}\)
```

Die Befehle `\overbrace` und `\underbrace` bewirken **waagrechte Klammern** über bzw. unter einem Ausdruck.

$\underbrace{a+b+\cdots+z}_{26}$

```
\(\underbrace{a+b+\cdots+z}_{26}\)
```

Um mathematische **Akzente** wie Pfeile oder Schlangen auf Variablen zu setzen, gibt es die in Tabelle 6 auf Seite 30 angeführten Befehle. Längere Tilden und Dacherln, die sich über mehrere (bis zu 3) Zeichen erstrecken können, erhält man mit `\widetilde` bzw. `\widehat`. Ableitungszeichen werden mit `'` (Apostroph) eingegeben.

$$y=x^{2} \qquad y'=2x \qquad y''=2$$

```
\begin{displaymath}
y=x^{2} \qquad
y'=2x   \qquad
y''=2
\end{displaymath}
```

Mathematische **Funktionen** werden in der Literatur üblicherweise nicht kursiv (wie die Namen von Variablen), sondern in „normaler“ Schrift dargestellt. LaTeX stellt die folgenden Befehle für mathematische Funktionen zur Verfügung:

`\arccos`	`\cos`	`\csc`	`\exp`	`\ker`	`\limsup`	`\min`	`\sinh`
`\arcsin`	`\cosh`	`\deg`	`\gcd`	`\lg`	`\ln`	`\Pr`	`\sup`
`\arctan`	`\cot`	`\det`	`\hom`	`\lim`	`\log`	`\sec`	`\tan`
`\arg`	`\coth`	`\dim`	`\inf`	`\liminf`	`\max`	`\sin`	`\tanh`

Für die Modulo-Funktion gibt es zwei verschiedene Befehle: `\bmod` für den binären Operator $a \bmod b$ und `\pmod{...}` für die Angabe in der Form $x \equiv a \pmod{b}$.

$$\lim_{x \to 0} \frac{\sin x}{x} = 1$$

```
\begin{displaymath}
\lim_{x \to 0} \frac{\sin x}{x}
=1
\end{displaymath}
```

Ein **Bruch** (fraction) wird mit dem Befehl `\frac{...}{...}` gesetzt. Für einfache Brüche kann man aber auch den Operator `/` verwenden.

$1\frac{1}{2}$ Stunden

$$\frac{x^{2}}{k+1} \qquad x^{\frac{2}{k+1}} \qquad x^{1/2}$$

```
\(1\frac{1}{2}\)~Stunden
\begin{displaymath}
\frac{ x^{2} }{ k+1 } \qquad
x^{ \frac{2}{k+1} } \qquad
x^{ 1/2 }
\end{displaymath}
```

Binomial-Koeffizienten können in der Form `{...\choose...}` gesetzt werden. Mit dem Befehl `\atop` erhält man das Gleiche ohne Klammern.

$$\binom{n}{k} \qquad \genfrac{}{}{0pt}{}{x}{y+2}$$

```
\begin{displaymath}
{ n \choose k } \qquad
{ x\atop y+2 }
\end{displaymath}
```

Das **Integralzeichen** wird mit `\int` eingegeben, das **Summenzeichen** mit `\sum`. Die obere und untere Grenze wird mit `^` bzw. `_` wie beim Hoch-/Tiefstellen angegeben.

Normalerweise werden die Grenzen neben das Integralzeichen gesetzt (um Platz zu sparen), durch Einfügen des Befehls `\limits` wird erreicht, daß die Grenzen oberhalb und unterhalb des Integralzeichens gesetzt werden.

Beim Summenzeichen hingegen werden die Grenzen bei der Angabe von `\nolimits` oder im laufenden Text neben das Summenzeichen gesetzt, ansonsten aber unter- und oberhalb.

$$\sum_{i=1}^{n} \qquad \int_{0}^{\frac{\pi}{2}} \qquad \int\limits_{-\infty}^{+\infty}$$

```
\begin{displaymath}
\sum_{i=1}^{n} \qquad
\int_{0}^{\frac{\pi}{2}} \qquad
\int \limits_{-\infty}^{+\infty}
\end{displaymath}
```

Für **Klammern** und andere Begrenzer gibt es in TEX viele verschiedene Symbole (z. B. $[\ \langle\ \|\ \ \updownarrow$). Runde und eckige Klammern können mit den entsprechenden Tasten eingegeben werden, geschwungene mit `\{`, die anderen mit speziellen Befehlen (z. B. `\updownarrow`).

Setzt man den Befehl `\left` vor öffnende Klammern und den Befehl `\right` vor schließende, so wird automatisch die richtige Größe gewählt.

$$1 + \left(\frac{1}{1-x^{2}} \right)^3$$

```
\begin{displaymath}
1 + \left( \frac{1}{ 1-x^{2} }
    \right) ^3
\end{displaymath}
```

In manchen Fällen möchte man die Größe der Klammern lieber selbst festlegen, dazu sind die Befehle `\bigl`, `\Bigl`, `\biggl` und `\Biggl` anstelle von `\left` und analog `\bigr` etc. anstelle von `\right` anzugeben.

$$\Bigl((x+1)(x-1) \Bigr)^{2}$$

```
\begin{displaymath}
\Bigl( (x+1) (x-1) \Bigr) ^{2}
\end{displaymath}
```

Um in Formeln **3 Punkte** (z. B. für $1, 2, \ldots, n$) auszugeben, gibt es die Befehle `\ldots` und `\cdots`. `\ldots` setzt die Punkte auf die Grundlinie (low), `\cdots` setzt sie in die Mitte der Zeilenhöhe (centered). Außerdem gibt es die Befehle `\vdots` für vertikal und `\ddots` für diagonal angeordnete Punkte.

$$x_{1},\ldots,x_{n} \qquad x_{1}+\cdots+x_{n}$$

```
\begin{displaymath}
x_{1},\ldots,x_{n} \qquad
x_{1}+\cdots+x_{n}
\end{displaymath}
```

4.3 Nebeneinander Setzen

Wenn man mit den von TEX gewählten **Abständen** innerhalb von Formeln nicht zufrieden ist, kann man sie mit expliziten Befehlen verändern. Die wichtigsten sind `\,` für einen sehr kleinen Abstand, `\␣` für einen mittleren, `\quad` und `\qquad` für große Abstände sowie `\!` für die Verkleinerung eines Abstands.

$$F_{n} = F_{n-1} + F_{n-2} \qquad n \ge 2$$

```
\begin{displaymath}
F_{n} = F_{n-1} + F_{n-2}
 \qquad n \ge 2
\end{displaymath}
```

$$\int\!\!\!\int_{D} dx\,dy \quad \textrm{statt} \quad \int\int_{D} dx dy$$

```
\begin{displaymath}
\int\!\!\!\int_{D} dx\,dy
\quad \textrm{statt} \quad
\int\int_{D} dx dy
\end{displaymath}
```

4.4 Übereinander Setzen

Für **Matrizen** u. ä. gibt es die `array`-Umgebung, die ähnlich wie die `tabular`-Umgebung funktioniert. Der Befehl `\\` trennt die Zeilen.

$$\mathbf{X} = \left(\begin{array}{ccc} x_{11} & x_{12} & \ldots \\ x_{21} & x_{22} & \ldots \\ \vdots & \vdots & \ddots \end{array} \right)$$

```
\begin{displaymath}
\mathbf{X} =
\left( \begin{array}{ccc}
x_{11} & x_{12} & \ldots \\
x_{21} & x_{22} & \ldots \\
\vdots & \vdots & \ddots
\end{array} \right)
\end{displaymath}
```

Für **mehrzeilige** Formeln oder Gleichungssysteme verwendet man die Umgebungen `eqnarray` und `eqnarray*` statt `equation`. Bei `eqnarray` erhält jede Zeile eine eigene Gleichungsnummer, bei `eqnarray*` wird ebenso wie bei `displaymath` *keine* Nummer hinzugefügt. Für Gleichungssysteme, die *eine* gemeinsame Nummer erhalten sollen, kann man eine `array`-Umgebung innerhalb der `equation`-Umgebung verwenden.

Die Umgebungen `eqnarray` und `eqnarray*` funktionieren wie eine 3-spaltige Tabelle der Form `{rcl}`, wobei die mittlere Spalte für das Gleichheits- oder Ungleichheitszeichen verwendet wird, nach dem die Zeilen ausgerichtet werden sollen. Der Befehl `\\` trennt die Zeilen.

$$\begin{array}{rcll} f(x) & = & \cos x & (5) \\ f'(x) & = & -\sin x & (6) \\ \int_{0}^{x} f(y)dy & = & \sin x & (7) \end{array}$$

```
\begin{eqnarray}
f(x) & = & \cos x        \\
f'(x) & = & -\sin x      \\
\int_{0}^{x} f(y)dy &
 = & \sin x
\end{eqnarray}
```

Zu lange Gleichungen werden von LaTeX *nicht* automatisch abgeteilt. Der Autor muß bestimmen, an welcher Stelle abgeteilt und wie weit eingerückt werden soll. Meistens verwendet man dafür eine der beiden folgenden Varianten:

$$\begin{array}{rcll} \sin x & = & x - \frac{x^{3}}{3!} + \frac{x^{5}}{5!} - & \\ & & {} - \frac{x^{7}}{7!} + \cdots & (8) \end{array}$$

```
\begin{eqnarray}
\sin x & = & x -\frac{x^{3}}{3!}
     +\frac{x^{5}}{5!} - {}
                    \nonumber\\
 & &{} -\frac{x^{7}}{7!} + \cdots
\end{eqnarray}
```

$$\begin{aligned} \cos x = 1 - \frac{x^2}{2!} + \\ + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots \qquad (9) \end{aligned}$$

```
\begin{eqnarray}
\lefteqn{ \cos x = 1
     -\frac{x^{2}}{2!} +{} }
                    \nonumber\\
 & &{} +\frac{x^{4}}{4!}
     -\frac{x^{6}}{6!} + \cdots
\end{eqnarray}
```

Der Befehl `\nonumber` bewirkt, daß an diese Stelle keine Gleichungsnummer gesetzt wird. Der Befehl `\lefteqn` ermöglicht Ausnahmen von der Spaltenaufteilung innerhalb `eqnarray`. Genauere Informationen enthält das *LaTeX-Handbuch* [1].

4.5 Liste der mathematischen Symbole

In den folgenden Tabellen sind alle Symbole angeführt, die standardmäßig im mathematischen Modus verwendet werden können. Die mit * versehenen Symbole werden in LaTeX 2_ε nur durch das Paket `latexsym` bereitgestellt. Bei vielen Installationen stehen mit den Paketen `amssymb`, `mathrsfs` oder `wasysym` weitere Zeichen zur Verfügung, näheres steht im *Local Guide* [7].

Tabelle 6: Mathematische Akzente

$\hat a$	`\hat a`	$\dot a$	`\dot a`	$\check a$	`\check a`
$\tilde a$	`\tilde a`	$\ddot a$	`\ddot a`	$\breve a$	`\breve a`
$\vec a$	`\vec a`	$\acute a$	`\acute a`	$\mathring a$	`\mathring a`
$\bar a$	`\bar a`	$\grave a$	`\grave a`		

Tabelle 7: Kleine griechische Buchstaben

α	`\alpha`	ι	`\iota`	ϱ	`\varrho`
β	`\beta`	κ	`\kappa`	σ	`\sigma`
γ	`\gamma`	λ	`\lambda`	ς	`\varsigma`
δ	`\delta`	μ	`\mu`	τ	`\tau`
ϵ	`\epsilon`	ν	`\nu`	υ	`\upsilon`
ε	`\varepsilon`	ξ	`\xi`	ϕ	`\phi`
ζ	`\zeta`	o	`o`	φ	`\varphi`
η	`\eta`	π	`\pi`	χ	`\chi`
θ	`\theta`	ϖ	`\varpi`	ψ	`\psi`
ϑ	`\vartheta`	ρ	`\rho`	ω	`\omega`

Tabelle 8: Große griechische Buchstaben

Γ	`\Gamma`	Ξ	`\Xi`	Φ	`\Phi`
Δ	`\Delta`	Π	`\Pi`	Ψ	`\Psi`
Θ	`\Theta`	Σ	`\Sigma`	Ω	`\Omega`
Λ	`\Lambda`	Υ	`\Upsilon`		

Tabelle 9: Verschiedene sonstige Symbole (* benötigt Paket `latexsym`)

$\aleph$	`\aleph`	$\prime$	`\prime`	$\forall$	`\forall`
$\hbar$	`\hbar`	$\emptyset$	`\emptyset`	$\exists$	`\exists`
$\imath$	`\imath`	∇	`\nabla`	$\neg$	`\neg`
$\jmath$	`\jmath`	$\surd$	`\surd`	$\flat$	`\flat`
ℓ	`\ell`	$\top$	`\top`	$\natural$	`\natural`
$\wp$	`\wp`	$\bot$	`\bot`	$\sharp$	`\sharp`
$\Re$	`\Re`	$\Diamond$	`\Diamond`*	$\clubsuit$	`\clubsuit`
$\Im$	`\Im`	$\Box$	`\Box`*	$\diamondsuit$	`\diamondsuit`
∂	`\partial`	$\triangle$	`\triangle`	$\heartsuit$	`\heartsuit`
∞	`\infty`	$\angle$	`\angle`	$\spadesuit$	`\spadesuit`
$\mho$	`\mho`*				

Tabelle 10: „Große“ Operatoren

$\textstyle\sum$	$\displaystyle\sum$	`\sum`	$\textstyle\bigcap$	$\displaystyle\bigcap$	`\bigcap`	$\textstyle\bigodot$	$\displaystyle\bigodot$	`\bigodot`
$\textstyle\prod$	$\displaystyle\prod$	`\prod`	$\textstyle\bigcup$	$\displaystyle\bigcup$	`\bigcup`	$\textstyle\bigotimes$	$\displaystyle\bigotimes$	`\bigotimes`
$\textstyle\coprod$	$\displaystyle\coprod$	`\coprod`	$\textstyle\bigsqcup$	$\displaystyle\bigsqcup$	`\bigsqcup`	$\textstyle\bigoplus$	$\displaystyle\bigoplus$	`\bigoplus`
$\textstyle\int$	$\displaystyle\int$	`\int`	$\textstyle\bigvee$	$\displaystyle\bigvee$	`\bigvee`	$\textstyle\biguplus$	$\displaystyle\biguplus$	`\biguplus`
$\textstyle\oint$	$\displaystyle\oint$	`\oint`	$\textstyle\bigwedge$	$\displaystyle\bigwedge$	`\bigwedge`			

Tabelle 11: Binäre Operatoren

$+$	`+`	$-$	`-`	$\div$	`\div`
$\pm$	`\pm`	$\cap$	`\cap`	$\vee$	`\vee`
$\mp$	`\mp`	$\cup$	`\cup`	$\wedge$	`\wedge`
$\setminus$	`\setminus`	$\uplus$	`\uplus`	$\oplus$	`\oplus`
$\cdot$	`\cdot`	$\sqcap$	`\sqcap`	$\ominus$	`\ominus`
$\times$	`\times`	$\sqcup$	`\sqcup`	$\otimes$	`\otimes`
$\ast$	`\ast`	$\triangleleft$	`\triangleleft`	$\oslash$	`\oslash`
$\star$	`\star`	$\triangleright$	`\triangleright`	$\odot$	`\odot`
$\diamond$	`\diamond`	$\lhd$	`\lhd`*	$\dagger$	`\dagger`
$\circ$	`\circ`	$\rhd$	`\rhd`*	$\ddagger$	`\ddagger`
$\bullet$	`\bullet`	$\unlhd$	`\unlhd`*	$\amalg$	`\amalg`
$\bigcirc$	`\bigcirc`	$\unrhd$	`\unrhd`*	$\wr$	`\wr`
$\bigtriangleup$	`\bigtriangleup`	$\bigtriangledown$	`\bigtriangledown`		

Tabelle 12: Relationen (* benötigt Paket `latexsym`)

$<$	`<`	$>$	`>`	$=$	`=`
$\leq$	`\leq`	$\geq$	`\geq`	$\equiv$	`\equiv`
$\prec$	`\prec`	$\succ$	`\succ`	$\sim$	`\sim`
$\preceq$	`\preceq`	$\succeq$	`\succeq`	$\simeq$	`\simeq`
$\ll$	`\ll`	$\gg$	`\gg`	$\asymp$	`\asymp`
$\subset$	`\subset`	$\supset$	`\supset`	$\approx$	`\approx`
$\subseteq$	`\subseteq`	$\supseteq$	`\supseteq`	$\cong$	`\cong`
$\sqsubseteq$	`\sqsubseteq`	$\sqsupseteq$	`\sqsupseteq`	$\bowtie$	`\bowtie`
$\sqsubset$	`sqsubset`*	$\sqsupset$	`sqsupset`*	$\Join$	`\Join`*
$\in$	`\in`	$\ni$	`\ni`	$\notin$	`\notin`
$\vdash$	`\vdash`	$\dashv$	`\dashv`	$\models$	`\models`
$\smile$	`\smile`	$\mid$	`\mid`	$\doteq$	`\doteq`
$\frown$	`\frown`	$\parallel$	`\parallel`	$\perp$	`\perp`
$:$	`:`	$\propto$	`\propto`		

Tabelle 13: Negierte Relationen

$\not<$	`\not<`	$\not>$	`\not>`	$\not=$	`\not=`
$\not\leq$	`\not\leq`	$\not\geq$	`\not\geq`	$\not\equiv$	`\not\equiv`
$\not\prec$	`\not\prec`	$\not\succ$	`\not\succ`	$\not\sim$	`\not\sim`
$\not\preceq$	`\not\preceq`	$\not\succeq$	`\not\succeq`	$\not\simeq$	`\not\simeq`
$\not\subset$	`\not\subset`	$\not\supset$	`\not\supset`	$\not\approx$	`\not\approx`
$\not\subseteq$	`\not\subseteq`	$\not\supseteq$	`\not\supseteq`	$\not\cong$	`\not\cong`
$\not\sqsubseteq$	`\not\sqsubseteq`	$\not\sqsupseteq$	`\not\sqsupseteq`	$\not\asymp$	`\not\asymp`

Tabelle 14: Pfeile (Vertikale Pfeile werden als Klammerungssymbole behandelt, alle anderen als Relationen. * benötigt Paket `latexsym`.)

$\leftarrow$	`\leftarrow`	$\longleftarrow$	`\longleftarrow`	$\uparrow$	`\uparrow`
$\Leftarrow$	`\Leftarrow`	$\Longleftarrow$	`\Longleftarrow`	$\Uparrow$	`\Uparrow`
$\rightarrow$	`\rightarrow`	$\longrightarrow$	`\longrightarrow`	$\downarrow$	`\downarrow`
$\Rightarrow$	`\Rightarrow`	$\Longrightarrow$	`\Longrightarrow`	$\Downarrow$	`\Downarrow`
$\leftrightarrow$	`\leftrightarrow`	$\longleftrightarrow$	`\longleft...`	$\updownarrow$	`\updownarrow`
$\Leftrightarrow$	`\Leftrightarrow`	$\Longleftrightarrow$	`\Longleft...`	$\Updownarrow$	`\Updownarrow`
$\mapsto$	`\mapsto`	$\longmapsto$	`\longmapsto`	$\nearrow$	`\nearrow`
$\hookleftarrow$	`\hookleftarrow`	$\hookrightarrow$	`\hookrightarrow`	$\searrow$	`\searrow`
$\leftharpoonup$	`\leftharpoonup`	$\rightharpoonup$	`\rightharpoonup`	$\swarrow$	`\swarrow`
$\leftharpoondown$	`\leftharpoondown`	$\rightharpoondown$	`\right...`	$\nwarrow$	`\nwarrow`
$\rightleftharpoons$	`\rightleftharpoons`			$\leadsto$	`\leadsto`*

Tabelle 15: Klammern

<table>
<tr><td>$($</td><td><code>(</code></td><td>$)$</td><td><code>)</code></td><td>$\lceil$</td><td><code>\lceil</code></td><td>$\rceil$</td><td><code>\rceil</code></td></tr>
<tr><td>$\langle$</td><td><code>\langle</code></td><td>$\rangle$</td><td><code>\rangle</code></td><td>$\lfloor$</td><td><code>\lfloor</code></td><td>$\rfloor$</td><td><code>\rfloor</code></td></tr>
<tr><td>$[$</td><td><code>[</code></td><td>$]$</td><td><code>]</code></td><td>$\{$</td><td><code>\{</code></td><td>$\}$</td><td><code>\}</code></td></tr>
<tr><td>$\vert$</td><td><code>|</code></td><td>$\Vert$</td><td><code>\|</code></td><td>$\backslash$</td><td><code>\backslash</code></td><td></td><td></td></tr>
</table>

5 Setzen von Bildern

Lädt man im Vorspann des Dokuments das Paket `graphics`, dann kann man Bilder, die mit einem anderen Programm erzeugt wurden, mit dem Befehl `\includegraphics` in das Dokument einfügen. Der Filename ist als Argument des Befehls anzugeben. Welche Arten von Grafikfiles verarbeitet werden können, hängt davon ab, mit welchem Drucker- oder Bildschirmtreiber das Schriftstück später ausgegeben oder angezeigt werden soll. Einzelheiten zum `graphics`-Paket finden Sie in [6], [11] und [12]; über die technischen Möglichkeiten Ihres speziellen LaTeX-Systems sollte [7] informieren.

Hier a.ps ist ein Bild.

```
Hier \includegraphics{a.ps} ist
ein Bild
```

Wird das Paket `graphics` mit der Option `[draft]` geladen. so erscheint anstelle des Bildes nur ein Rahmen entsprechend der tatsächlichen Bildgröße mit dem Namen des Grafikfiles, was für Probeausdrucke nützlich ist – und hier nur sicherstellt, daß die vorliegende Kurzanleitung wirklich mit jeder Treibersoftware wiedergegeben werden kann.

6 Seitenaufbau

6.1 Kopf- und Fußzeilen

Der Inhalt von Kopf- und Fußzeilen kann mit dem Befehl

`\pagestyle{`*style*`}`

festgelegt werden:

Mit `\pagestyle{plain}` steht die Seitennummer zentriert in der Fußzeile; das ist die Voreinstellung und braucht normalerweise nicht explizit angegeben zu werden. Mit dem Stil `headings` stehen Kapitel-Überschrift und Seitennummer in der Kopfzeile. Mit `empty` sind Kopf- und Fußzeile leer. Der Befehl

`\thispagestyle{`*style*`}`

gilt entsprechend nur für die aktuelle Seite. Einige Befehle, wie etwa `\chapter`, ändern den Stil der aktuellen Seite. Diese Änderungen kann man durch einen nachfolgenden `\thispagestyle`-Befehl aufheben.

Im *LaTeX-Handbuch* [1] ist angegeben, wie man das Aussehen der Kopf- und Fußzeilen außerdem mit dem Seitenstil `myheadings` und den Befehlen `\markboth`, `\markright` und `\pagenumbering` beeinflussen kann.

6.2 Gleitobjekte

Große Bilder und lange Tabellen lassen sich nicht immer genau dort unterbringen, wo sie inhaltlich hingehören, weil sie nicht mehr vollständig auf die aktuelle Seite passen, aber auch nicht durch einen Seitenwechsel zerrissen werden sollen. Um solche Strukturen automatisch an eine geeignete Stelle „gleiten“ zu lassen, kennt LaTeX die beiden Umgebungen `figure` und `table`.

6.2.1 Abbildungen (figure)

Diese Umgebung ist für die Behandlung von Abbildungen gedacht. Tatsächlich spielt es aber keine Rolle, *wie* diese erzeugt wurden: Alles, was zwischen `\begin{figure}` und `\end{figure}` steht, wird automatisch an eine Stelle gesetzt, wo es komplett hinpaßt, ohne durch einen Seitenwechsel zerrissen zu werden.

Mit `\caption{...}` setzt man die Bezeichnung der Abbildung. Dabei ist nur der Text anzugeben, das Wort „Abbildung“ und die fortlaufende Nummer werden von LaTeX hinzugefügt. Bei Abbildungen ist es allgemein üblich, die Bezeichnung *unter* das Bild zu setzen. Mit `\label` und `\ref` kann man die Nummer der Abbildung im Text ansprechen, mit `\pageref` ihre Seitenzahl. Der Befehl `\label` muß dabei *nach* dem `\caption`-Befehl stehen, sonst stimmt die Numerierung nicht!

Im folgenden Beispiel wird einfach mit dem Befehl `\vspace` (siehe Abschnitt 8.1.3) leerer Raum für ein später einzusetzendes Bild gelassen:

Abbildung 6 auf S. 36 zeigt ein Beispiel aus der Minimal art.

```
Abbildung~\ref{weiss} auf
S.~\pageref{weiss} zeigt
ein Beispiel aus der
Minimal art.
\begin{figure}[tb]
\vspace{6cm}
\caption{Landschaft im
Nebel} \label{weiss}
\end{figure}
```

LaTeX kann eine Abbildung nach verschiedenen Kriterien plazieren: `h` „here“ (hier), `t` „top“ (oben auf der Seite), `b` „bottom“ (unten auf der Seite) oder `p` „page“ (eigene Seite für Abbildungen).

Die Parameter in den eckigen Klammern, die wahlweise angegeben werden können, dienen dazu, die Plazierung der Abbildung auf die angegebenen Orte *einzuschränken.* Durch Angabe von z. B. `tb` wird LaTeX angewiesen, nur eine Plazierung oben oder unten auf der Seite zu versuchen, je nachdem, wo *zuerst* eine passende Stelle gefunden wird. Werden keine Parameter (und keine eckigen Klammern!) angegeben, ist die Voreinstellung `tbp`, also ohne `h`.

Eine Plazierungsbeschränkung *nur* auf `[h]` ist unsinnig; sie würde das „Gleiten“ ja gerade verhindern. Wenn der Platz „hier“ nicht ausreicht, verschiebt LaTeX dann die Abbildung mindestens bis zum Anfang der nächsten Seite, so als hätte man `[ht]` angegeben.

Abbildung 6: Landschaft im Nebel

Eine Abbildung, die nicht plaziert werden konnte, wird von LaTeX immer weiter nach hinten verschoben (und schiebt alle weiteren Abbildungen vor sich her!), bis ein neues Kapitel beginnt, das Dokument zu Ende ist, oder der Befehl `\clearpage` eingegeben wird.

Es gibt noch einen weiteren Plazierungsparameter, `!` (bang), der LaTeX anweist, gewisse eingebaute Beschränkungen zu ignorieren, z. B., daß bei der Plazierung gemäß `h`, `t` oder `b` ein Mindestanteil der Seite für normalen Text übrig bleiben muß. „Bang“ muß immer zusammen mit mindestens einem der vier anderen Parameter benutzt werden.

6.2.2 Tabellen (table)

Damit Tabellen nicht auf einen Seitenwechsel fallen, können sie, analog zu Abbildungen, zwischen `\begin{table}` und `\end{table}` gesetzt werden. Die Befehle `\caption`, `\label`, `\ref` und `\pageref` wirken entsprechend. Hier sind beide möglichen Konventionen verbreitet: Die Bezeichnung wird entweder immer *über* oder immer *unter* die Tabelle gesetzt.

Auch hier gilt, daß in der `table`-Umgebung beliebiger Text stehen darf; die Tabelle muß nicht zwangsläufig durch die `tabular`-Umgebung erzeugt worden sein. Der Unterschied zu `figure` besteht nur darin, daß die Bezeichnung mit dem Wort „Tabelle“ versehen wird, und daß die Tabellen unabhängig von den Abbildungen numeriert werden.

7 Schriften

Normalerweise wählt LaTeX die Größe und den Stil der Schrift aufgrund der Befehle aus, die die logische Struktur des Textes angeben: Überschriften, Fußnoten, Hervorhebungen usw. Im folgenden werden Befehle und Makropakete beschrieben, mit denen die Schrift auch explizit beeinflußt werden kann. Ausführlichere Erläuterungen zum Umgang mit Schriften in LaTeX findet man im *LaTeX-Begleiter* [3] und in der Online-Dokumentation [9].

7.1 Schriftgrößen

Die in der Tabelle 16 angeführten Befehlen wechseln die Schriftgröße. Sie spezifizieren die Größe relativ zu der von `\documentclass` festgelegten Grundschrift. Ihr Wirkung reicht bis zum Ende der aktuellen Gruppe oder Umgebung.

Tabelle 16: Schriftgrößen

`\tiny`	winzig kleine Schrift
`\scriptsize`	sehr kleine Schrift (wie Indizes)
`\footnotesize`	kleine Schrift (wie Fußnoten)
`\small`	kleine Schrift
`\normalsize`	normale Schrift
`\large`	große Schrift
`\Large`	größere Schrift
`\LARGE`	sehr große Schrift
`\huge`	riesig groß
`\Huge`	gigantisch

Die Größen-Befehle verändern auch die Zeilenabstände auf die jeweils passenden Werte – aber nur, wenn die Leerzeile, die den Absatz beendet, innerhalb des Gültigkeitsbereichs des Größen-Befehls liegt:

zu enger
Abstand

```
{\Large zu enger \\
Abstand}\par
```

richtiger
Abstand

```
{\Large richtiger\\
Abstand\par}
```

Für korrekte Zeilenabstände darf die schließende geschwungene Klammer also nicht zu früh kommen, sondern erst nach einer Leerzeile oder einem explizit mit dem Befehl `\par` eingefügten Absatzende.

7.2 Schriftstil

Der Schriftstil wird in LaTeX durch 3 Merkmale definiert:

Familie Standardmäßig stehen 3 Familien zur Wahl: „roman“ (Antiqua), „sans serif“ (Serifenlose) und „typewriter“ (Schreibmaschinenschrift).

Serie Die Serie gibt Stärke und Laufweite der Schrift an: „medium“ (normale Schrift), „boldface extended“ (fett und breiter).

Form Die Form der Buchstaben: „upright“ (aufrecht), „slanted“ (geneigt), „italic“ (kursiv), „caps and small caps“ (Kapitälchen).

Tabelle 17 zeigt die Befehle, mit denen diese Attribute explizit beeinflußt werden können. Die Befehle der Form `\text...` setzen nur ihr Argument im gewünschten Stil. Zu jedem dieser Befehle ist ein Gegenstück angegeben, das von seinem Auftreten an bis zum Ende der laufenden Gruppe oder Umgebung wirkt.

Zu beachten ist, daß Wörter in Schreibmaschinenschrift nicht automatisch getrennt werden.

Tabelle 17: Schriftstile

`\textrm`{*text*}	`\rmfamily`	Antiqua
`\textsf`{*text*}	`\sffamily`	Serifenlose
`\texttt`{*text*}	`\ttfamily`	`Maschinenschrift`
`\textmd`{*text*}	`\mdseries`	normal
`\textbf`{*text*}	`\bfseries`	**fett, breiter laufend**
`\textup`{*text*}	`\upshape`	aufrecht
`\textsl`{*text*}	`\slshape`	*geneigt*
`\textit`{*text*}	`\itshape`	*kursiv*
`\textsc`{*text*}	`\scshape`	KAPITÄLCHEN
`\textnormal`{*text*}	`\normalfont`	Die Grundschrift des Dokuments

Die Befehle für Familie, Serie und Form können untereinander und mit den Größen-Befehlen kombiniert werden; allerdings muß nicht jede mögliche Kombination tatsächlich als reale Schrift (Font) zur Verfügung stehen.

Die kleinen **fetten** Römer beherrschten das ganze große *Italien.*

plakativ

```
{\small Die kleinen
\textbf{fetten} R"omer
beherrschten }{\large das
ganze gro"se \textit{Italien}.}
{\Large\sffamily\slshape plakativ}
```

Je *weniger* verschiedene Schriftarten man verwendet, desto lesbarer und schöner wird das Schriftstück!

7.3 Andere Schriftfamilien

Mit den im vorigen Abschnitt eingeführten Befehlen kann man nicht beeinflussen, welche Schriftfamilien tatsächlich als Antiqua, Serifenlose und Maschinenschrift benutzt werden. LaTeX verwendet als Voreinstellung die sog. Computer-Modern-Schriftfamilien (CM), siehe Tabelle 18; der Stil der mathematischen Zeichensätze paßt dabei zu CM Roman.

Will man andere Schriften benutzen, dann ist der einfachste Weg das Laden eines Pakets, das eine oder mehrere dieser Schriftfamilien komplett ersetzt. Tabelle 18 führt einige derartige Pakete auf, die allerdings nicht mit jeder LaTeX-Installation verfügbar sein müssen. Bei den Schriftfamilien „Times“, „Palatino“, „Helvetica“ und „Courier“ handelt es sich um Type-1-Fonts. Ihre Verwendung setzt normalerweise voraus, daß die von LaTeX erzeugte dvi-Datei zunächst in das PostScript-Format umgewandelt wird, bevor sie angezeigt oder ausgedruckt werden kann. In der Beschreibung [7] Ihres LaTeX-Systems sollte angegeben sein, ob und, wenn ja, wie das unterstützt wird. Näheres zu PostScript-Schriften finden Sie in [3], [4] und [11].

Tabelle 18: Pakete für alternative Schriftfamilien (Eine leere Tabellenspalte bedeutet, daß das Paket die betreffende Schriftfamilie nicht verändert; * kennzeichnet die jeweils als Grundschrift eingestellte Familie.)

Paket	Antiqua	Serifenlose	Schreibmaschine	math. Formeln
(keines)	CM Roman *	CM Sans Serif	CM Typewriter	$\approx$ CM Roman
`ccfonts`	Concrete *			$\approx$ Concrete
`cmbright`		CM Bright *	CM Typewriter Light	$\approx$ CM Bright
`pandora`	Pandora Roman *	Pandora Sans Serif		
`mathptmx`	Times *			$\approx$ Times
`mathpple`	Palatino *			$\approx$ Palatino
`helvet`		Helvetica		
`courier`			Courier	

7.4 Die europäischen Schriften

LaTeX verwendet standardmäßig Schriften mit einem Umfang von 128 Zeichen. Umlaute oder akzentuierte Buchstaben sind darin nicht enthalten; sie werden jeweils aus dem Grundsymbol und dem Akzent zusammengesetzt. Seit 1997 existieren zusätzlich sogenannte „europäische“ Schriften. Sie enthalten 256 Zeichen, welche fast alle europäischen Sprachen abdecken, d. h., jedes benötigte Zeichen ist vorgefertigt in ihnen enthalten.

Dies hat nicht nur eine höhere typographische Qualität zur Folge; aufgrund der inneren Arbeitsweise von TEX entfallen damit auch die Einschränkungen im Zusammenhang mit der Silbentrennung, die im Abschnitt 3.2.2 erwähnt wurden: Wörter mit Umlauten werden nun besser getrennt, und im Argument des Befehls `\hyphenation` dürfen auch Umlaute und das scharfe s stehen. Weiterhin sind die Unterschneidungen im Vergleich zu den amerikanischen TEX-Originalschriften stark verbessert und nun auch auf häufige Buchstabenpaarungen in nicht-englischen Sprachen optimiert.

Die europäischen Schriften bestehen aus zwei Teilen: Die T1-Schriften enthalten Buchstaben, ASCII-Zeichen sowie verschiedene Anführungszeichen und Striche, während die TS1-Schriften zusätzliche Textsymbole bereitstellen.

LATEX wird veranlaßt, T1-Schriften zu verwenden, indem man das Paket `fontenc` mit der Option `T1` lädt:

```
\usepackage[T1]{fontenc}
```

Das Paket `textcomp` ermöglicht den Zugriff auf die Textsymbole:

```
\usepackage{textcomp}
```

Welche zusätzlichen Zeichen mit den T1-Schriften bereitgestellt werden, ist in [8] zusammengefaßt; Anhang A der vorliegenden Kurzbeschreibung enthält eine Liste aller TS1-Textsymbole. Einige der Textsymbole sind auch ohne das Paket `textcomp` verfügbar, siehe Abschnitt 3.4.5, dann aber nicht immer in einem zur laufenden Schrift passenden Stil.

Alle Schriftfamilien der Tabelle 18, mit Ausnahme von `pandora`, stehen sowohl mit dem standardmäßigen Zeichenvorrat als auch in Form der europäischen Schriften zur Verfügung. In den PostScript-Schriften fehlen allerdings einzelne Zeichen, besonders aus den Textsymbolen.

8 Spezialitäten

Das komplette Menü der Spezialitäten, die von LaTeX serviert werden, ist im *LaTeX-Handbuch* [1] beschrieben. Hier soll nur auf einige besondere „Zuckerln“ hingewiesen werden.

8.1 Abstände

8.1.1 Zeilenabstand

Um in einem Schriftstück größere Zeilenabstände zu verwenden, als es in der Dokumentklasse vorgesehen ist, gibt es in LaTeX den Befehl `\linespread`:

> für „eineinhalbzeilige“ Ausgabe:
> `\linespread{1.3}`
>
> für „doppelzeilige“ Ausgabe:
> `\linespread{1.6}`

8.1.2 Spezielle horizontale Abstände

Die Abstände zwischen Wörtern und Sätzen werden von LaTeX automatisch gesetzt. Sonstigen horizontalen Abstand kann man mit dem Befehl

> `\hspace{`*länge*`}`

einfügen. Wenn der Abstand auch am Beginn oder Ende einer Zeile erhalten bleiben soll, muß `\hspace*` statt `\hspace` geschrieben werden. Die Längenangabe besteht im einfachsten Fall aus einer Zahl und einer Einheit. Die wichtigsten Einheiten sind in Tabelle 19 angeführt. Die Befehle in Tabelle 20 sind Abkürzungen zum Einfügen besonderer horizontaler Abstände. Der Befehl `\hfill` kann dazu dienen, einen vorgegebenen Platz auszufüllen.

Schafft mir Raum!	`Schafft mir\hspace{1.5cm}Raum! \\`
$\triangleleft$ $\triangleright$	`\(\triangleleft\)\hfill \(\triangleright\)`

Tabelle 19: Einheiten für Längenangaben

`mm`	Millimeter
`cm`	Zentimeter = 10 mm
`in`	inch = 25.4 mm
`pt`	point = (1/72.27) in ≈ 0.351 mm
`bp`	big point = (1/72) in ≈ 0.353 mm
`em`	Geviert (doppelte Breite einer Ziffer der aktuellen Schrift)
`ex`	Höhe des Buchstabens x der aktuellen Schrift

Tabelle 20: Befehle für horizontale Abstände

`\,`	ein sehr kleiner Abstand (siehe auch Abschnitt 3.3)
`\enspace`	so breit wie eine Ziffer
`\quad`	so breit, wie ein Buchstabe hoch ist („weißes Quadrat“)
`\qquad`	doppelt so breit wie ein `\quad`
`\hfill`	ein Abstand, der sich von 0 bis ∞ ausdehnen kann.

Tabelle 21: Befehle für vertikale Abstände

`\smallskip`	etwa $^1/_4$ Zeile
`\medskip`	etwa $^1/_2$ Zeile
`\bigskip`	etwa 1 Zeile
`\vfill`	ein Abstand, der sich von 0 bis ∞ ausdehnen kann

8.1.3 Spezielle vertikale Abstände

Die Abstände zwischen Absätzen, Kapiteln usw. werden von LaTeX automatisch bestimmt. In Spezialfällen kann man zusätzlichen Abstand *zwischen zwei Absätzen* mit dem Befehl

`\vspace{`*länge*`}`

bewirken. Dieser Befehl sollte immer zwischen zwei Leerzeilen angegeben werden. Wenn der Abstand auch am Beginn oder Ende einer Seite erhalten bleiben soll, muß `\vspace*` statt `\vspace` geschrieben werden. Die Befehle in Tabelle 21 sind Abkürzungen für bestimmte vertikale Abstände. Der Befehl `\vfill` in Verbindung mit `\newpage` kann dazu dienen, Text an den unteren Rand einer Seite zu setzen oder vertikal zu zentrieren. Beispielsweise enthält der Quelltext für die zweite Seite der vorliegenden Beschreibung:

```
\vfill

Dieses Dokument wurde mit \LaTeX{} gesetzt.
...
\newpage
```

Zusätzlichen Abstand zwischen zwei Zeilen *innerhalb* eines Absatzes oder einer Tabelle erreicht man mit dem Befehl `\\[`*länge*`]`.

Albano Cesara
Lindenallee 10

95632 Pestitz

```
Albano Cesara \\
Lindenallee 10 \\[1.5ex]
95632 Pestitz
```

8.2 Briefe

Mit der Dokumentklasse `letter` kann man zwischen `\begin{document}` und `\end{document}` einen oder mehrere Briefe schreiben. Abbildung 7 enthält ein Beispiel für einen Brief.

```
\documentclass[12pt,a4paper]{letter}
\usepackage[latin1]{inputenc}
\usepackage{german}
\address{EDV-Zentrum der TU Wien \\
  Abt. Digitalrechenanlage \\
  Wiedner Hauptstraße 8--10 \\
  A-1040 Wien}
\signature{Dr. Hubert Partl}
\begin{document}
\begin{letter}{Frau Mag. Elisabeth Schlegl \\
  EDV-Zentrum der Karl-Franzens-Universität \\
  Attemsgasse 25/II \\
  \textbf{A-8010 Graz}}
\opening{Liebe Frau Schlegl,}
herzlichen Dank für die Zusendung ...

... in etwa 2--3~Wochen fertig zu sein.
\closing{Mit freundlichen Grüßen}
\end{letter}
\end{document}
```

Abbildung 7: Brief von H. P. an E. S.

Mit dem Befehl `\address` definiert man die Adresse des Absenders. `\begin{letter}{...}` beginnt einen Brief an den im Parameter angegebenen Empfänger. `\opening{...}` schreibt die Anrede und `\closing{...}` den abschließenden Gruß, an den automatisch die eingangs mit `\signature` vereinbarte Unterschrift angefügt wird. `\end{letter}` beendet den jeweiligen Brief.

Das von der Dokumentklasse `letter` bewirkte Layout der Briefe orientiert sich an amerikanischen Gepflogenheiten. Mit vielen LaTeX-Systemen ist die Klasse `dinbrief` verfügbar; sie setzt die Briefe in einer Anordnung gemäß DIN 676, die für die Verwendung von A4-Bogen in Fensterkuverts geeignet ist. Der *Local Guide* [7] sollte Auskunft über diese oder andere Alternativen zu `letter` geben.

8.3 Literaturangaben

Mit der `thebibliography`-Umgebung kann man ein Literaturverzeichnis drukken. Darin beginnt jede Literaturangabe mit `\bibitem`. Als Parameter wird ein Name vereinbart, unter dem die Literaturstelle im Text zitiert werden kann, und dann folgt der Text der Literaturangabe. Die Numerierung erfolgt automatisch. Der Parameter bei `\begin{thebibliography}` gibt die maximale Breite dieser Nummernangabe an, also z. B. `{99}` für maximal zweistellige Nummern.

Im Text zitiert man die Literaturstelle dann mit dem Befehl `\cite` und dem vereinbarten Namen als Argument.

Partl [1] hat vorgeschlagen, daß ...

Literatur

[1] H. Partl: *German TEX,* TUGboat Vol. 9, No. 1 (1988)

```
Partl~\cite{pa} hat
vorgeschlagen ...

\begin{thebibliography}{99}
\bibitem{pa}
H.~Partl: \textit{German \TeX,}
TUGboat Vol.~9, No.~1 (1988)
\end{thebibliography}
```

8.4 Zerbrechliche Befehle

Manche LATEX-Befehle „verfrachten“ ihre Argumente an eine andere Stelle im Text. Beispielsweise kann das Argument von `\section` auch im Inhaltsverzeichnis und möglicherweise in der Kopfzeile auftauchen.

Bestimmte Befehle „überstehen“ diesen Transport nicht, wenn sie ohne besondere Maßnahmen in einem solchen „beweglichen Argument“ auftreten. Derartige Befehle heißen „zerbrechlich“. Damit sie dennoch innnerhalb von beweglichen Argumenten benutzt werden dürfen, muß man ihnen einfach den Befehl `\protect` voranstellen.

Zerbrechlich sind insbesondere alle Befehle, die ein optionales Argument kennen, also auch `\\` (sic!), außerdem die Befehle `\(`, `\)` und `\footnote`.

Bewegliche Argumente haben, neben den Gliederungsbefehlen, auch der Befehl `\caption` und die Umgebung `letter`.

A Mit dem Paket `textcomp` verfügbare Symbole

‚	\textquotestraightbase*	„	\textquotestraightdblbase*
—	\texttwelveudash*	—	\textthreequartersemdash*
←	\textleftarrow	→	\textrightarrow
␢	\textblank	$	\$*
'	\textquotesingle*	∗	\textasteriskcentered*
⹀	\textdblhyphen	⁄	\textfractionsolidus*
〈	\textlangle	−	\textminus*
〉	\textrangle	℧	\textmho
◯	\textbigcircle	Ω	\textohm
⟦	\textlbrackdbl	⟧	\textrbrackdbl
↑	\textuparrow	↓	\textdownarrow
`	\textasciigrave*	⋆	\textborn
⚮	\textdivorced	†	\textdied
❦	\textleaf	⚭	\textmarried
♪	\textmusicalnote	˷	\texttildelow*
⹀	\textdblhyphenchar	˘	\textasciibreve*
ˇ	\textasciicaron*	˝	\textacutedbl*
̏	\textgravedbl*	†	\dag*
‡	\ddag*	‖	\textbardbl*
‰	\textperthousand*	•	\textbullet*
℃	\textcelsius*	$	\textdollaroldstyle
¢	\textcentoldstyle	ƒ	\textflorin*
₡	\textcolonmonetary	₩	\textwon
₦	\textnaira	₲	\textguarani
₱	\textpeso	₤	\textlira
℞	\textrecipe	‽	\textinterrobang
⸘	\textinterrobangdown	₫	\textdong
™	\texttrademark*	‱	\textpertenthousand
¶	\textpilcrow	฿	\textbaht
№	\textnumero	⁒	\textdiscount
℮	\textestimated	◦	\textopenbullet
℠	\textservicemark	⁅	\textlquill
⁆	\textrquill	¢	\textcent*
£	\pounds*	¤	\textcurrency*
¥	\textyen*	¦	\textbrokenbar*
§	\S*	¨	\textasciidieresis*
©	\copyright*	ª	\textordfeminine*
🄯	\textcopyleft	¬	\textlnot*
℗	\textcircledP	®	\textregistered*
¯	\textasciimacron*	°	\textdegree*
±	\textpm*	^{2}	\texttwosuperior
^{3}	\textthreesuperior	´	\textasciiacute*
µ	\textmu*	¶	\P*
·	\textperiodcentered*	※	\textreferencemark
^{1}	\textonesuperior	º	\textordmasculine*
√	\textsurd	¼	\textonequarter
½	\textonehalf	¾	\textthreequarters
€	\textsf{\texteuro}	×	\texttimes*
÷	\textdiv*		

In PostScript-Schriften, die nicht speziell für die Verwendung mit TeX entworfen wurden, gibt es es normalerweise nur die mit * markierten Zeichen.

Literatur

[1] L. Lamport: *Das LaTeX-Handbuch.* Addison-Wesley Deutschland (1995). Deutsche Übersetzung von [2].

[2] L. Lamport: *LaTeX, A Document Preparation System.* Addison-Wesley, 2. Aufl. (1994).

[3] M. Goossens, F. Mittelbach und A. Samarin: *Der LaTeX-Begleiter.* Addison Wesley Longman, 2. korr. Nachdruck (1996). Deutsche Übersetzung von [4].

[4] M. Goossens, F. Mittelbach und A. Samarin: *The LaTeX Companion.* Addison-Wesley (1994).

[5] M. Goossens, F. Mittelbach und A. Samarin: *Higher Mathematics.* `<ftp://ftp.dante.de/tex-archive/info/companion-rev/ch8.pdf>` (1998). Aktualisierte Fassung von Kapitel 8 aus [4].

[6] M. Goossens, S. Rahtz und F. Mittelbach: *The LaTeX Graphics Companion.* Addison Wesley Longman (1997).

[7] Zu jedem installierten LaTeX-System sollte ein *LaTeX Local Guide* vorhanden sein, in dem alle für dieses System spezifischen Angaben – z. B. die für den Aufruf der Programme notwendigen Befehle und die zur Verfügung stehenden Dokumentklassen, Pakete und Schriften – angeführt sind.

[8] LaTeX3 Project Team (Hrsg.): *LaTeX 2ε for authors.* Bestandteil der Online-Dokumentation von LaTeX, Datei `usrguide.tex`. Aktuelle Änderungen und Ergänzungen sowie die Unterschiede zum früheren LaTeX 2.09 sind hier dokumentiert.

[9] LaTeX3 Project Team (Hrsg.): *LaTeX 2ε font selection.* Bestandteil der Online-Dokumentation von LaTeX, Datei `fntguide.tex`.

[10] LaTeX3 Project Team (Hrsg.): *LaTeX 2ε for class and package writers.* Bestandteil der Online-Dokumentation von LaTeX, Datei `clsguide.tex`.

[11] D. P. Carlisle: *Packages in the 'graphics' bundle.* Online-Dokumentation des `graphics`-Pakets, Datei `grfguide.ps`.

[12] K. Reckdahl: *Using Imported Graphics in LaTeX 2ε.* `<ftp://ftp.dante.de/tex-archive/info/epslatex.ps>`

[13] B. Raichle: *Kurzbeschreibung – `german.sty`.* `<ftp://ftp.dante.de/tex-archive/language/german/gerdoc.tex>`

[14] D. E. Knuth: *Computers & Typesetting, Vol. A: The TeX Book.* Addison-Wesley (1991).

[15] N. Schwarz: *Einführung in TeX– incl. Version 3.0.* Oldenbourg, 3. Aufl. (1991).

[16] H. Partl: *German TeX.* TUGboat Vol. 9, No. 1 (1988).

[17] H. Partl und A. Kielhorn: *Layout-Änderungen mit LaTeX.* EDV-Zentrum der Technischen Universität Wien, `<ftp://ftp.dante.de/tex-archive/macros/latex/contrib/supported/refman/>` (1996).

[18] D. F. Langmyhr: *How to make your own document style in LaTeX2ε.* In: *Proceedings of the Eighth European TeX Conference* (1994).

[19] A. Reichert: *Typografie – Gestaltung einer Beispielklasse.* `<ftp://ftp.dante.de/info/german/typografie/>` (1999).