

Erfahrungen mit LaTeX  
an der Schloss-Schule Ilvesheim

Rolf Bernd

25. September 2002

# Inhaltsverzeichnis

<b>1</b>	<b>Warum LaTeX</b>	<b>3</b>
<b>2</b>	<b>Winword - vom Editor zur TexShell</b>	<b>5</b>
2.1	Vorbereitungen in Word . . . . .	5
2.2	Umlaute und Sonderzeichen . . . . .	6
2.3	Umschließende Befehle eingeben . . . . .	6
<b>3</b>	<b>Die Menübefehle unserer TeXShell mit Beispielen</b>	<b>7</b>
3.1	Das Seitenlayout . . . . .	7
3.2	Das Untermenü Einfügen (Alt-S-E) . . . . .	7
3.2.1	Titelseite (Alt-S-E-T) . . . . .	7
3.2.2	Inhaltsverzeichnis (Alt-S-E-V) . . . . .	7
3.2.3	Fußnoten (Alt-S-E-F) . . . . .	7
3.2.4	Zeilenumbruch (Alt-S-E-Z) . . . . .	8
3.2.5	Seitenumbruch (Alt-S-E-S) . . . . .	8
3.3	Überschriften (Alt-S-Ü) . . . . .	8
3.4	Absatzformate . . . . .	8
3.4.1	Fließtext . . . . .	8
3.4.2	Absatz linksbündig (Alt-S-A-L) . . . . .	9
3.4.3	Absatz rechtsbündig (Alt-S-A-R) . . . . .	9
3.4.4	Absatz zentriert (Alt-S-A-Z) . . . . .	9
3.4.5	Absatz eingerückt (Alt-S-A-E) . . . . .	9
3.4.6	Absatz nummeriert (Alt-S-A-N) . . . . .	9
3.4.7	Absatz Spiegelstrich (Alt-S-A-S) . . . . .	10
3.5	Tabellen (Alt-S-A-T) . . . . .	10
3.5.1	Tabellen mit mehrzeiligen Spalten . . . . .	11
3.5.2	Kombinationen von Tabellenstrukturen . . . . .	11
3.6	Schriftarten, Schriftattribute und Schriftgrößen . . . . .	12
3.6.1	Schriftarten . . . . .	12
3.6.2	Schriftattribute . . . . .	12
3.6.3	Schriftgrößen . . . . .	12
3.7	Mathematik . . . . .	13
3.7.1	Mathetext oder abgesetzte Formeln (Alt-S-M-X oder Alt-S-M-F) . . . . .	13
3.7.2	Gleichungsfelder (Alt-S-M-G) . . . . .	13
3.7.3	Zeilenumbrüche und Kommentare im Quelltext . . . . .	14
3.8	Übersicht der Befehle im Menü TeXShell in der Vorlagen-Datei latex.dot . . . . .	15

3.8.1	Dokumente erstellen . . . . .	15
3.8.2	Überschriften . . . . .	15
3.8.3	Absatzformate . . . . .	15
3.8.4	Allgemeine Elemente im Dokument . . . . .	16
3.8.5	Mathematische Befehle . . . . .	16
3.8.6	Schriftarten . . . . .	17
3.8.7	Schriftattribute . . . . .	17
3.8.8	Schriftgrößen . . . . .	17
<b>4</b>	<b>Das griechische Alphabet</b>	<b>18</b>
4.0.9	Zwei verschiedene Zeichen gibt es bei folgenden Buchstaben . . . . .	18
4.0.10	Bei folgenden Buchstaben gibt es keinen eigenen Großbuchstaben . . . . .	18

# 1 Warum LaTeX

Ich habe keine Ahnung von Mathematik. Dennoch war der Hauptgrund für meine Beschäftigung mit LaTeX die Möglichkeit des Formelsatzes. Beim Erstellen von Texten für blinde Schüler, die an allgemeinen Schulen integriert waren, war es problemlos, Texte in Deutsch, Englisch, Französisch, Spanisch und Latein zur Verfügung zu stellen, da die Schüler alle am Computer in Eurobraille arbeiteten.

In Physik und Mathematik gab es aber riesige Probleme. Manche arbeiteten mit der Stuttgarter Mathematikschrift, manche mit der Karlsruher, andere benutzten die Syntax von Mathematikprogrammen wie z.B. Derive. Daneben gab es einen Wildwuchs von individuell zwischen einzelnen Schülern und ihren Mathematiklehrern vereinbarten Codes. Selbst wer die Stuttgarter Mathematikschrift benutzte, fand es oft lästig, jeweils den notwendigen Zeichensatz in die Braillezeile und den Brailledrucker zu übertragen. Diese Schüler benutzten die Stuttgarter Mathematikschrift mit dem Standard ASCII Zeichensatz. Dieser Zustand war unbefriedigend.

Bei einer Diskussion zu dieser Problematik hörte ich von Uli Kalina zum ersten Mal bewußt das Wort LaTeX. Seine lange vorher veröffentlichten Anregungen zu diesem Thema hatte ich entweder nicht zur Kenntnis genommen oder ihre Bedeutung nicht begriffen. Erst die Arbeit im Textservice für blinde Schüler an allgemeinen Schulen weckte meine Aufmerksamkeit. Also kauften wir ein Buch über LaTeX und begannen, an unserer Schule mit LaTeX zu experimentieren.

Es klingt vielleicht absurd, aber den größten Vorteil von LaTeX als Mathematikschrift für Blinde sehe ich darin, dass es weder eine Mathematikschrift, noch eine Schrift für Blinde ist. LaTeX ist zunächst einmal ein weltweit verbreitetes Satzsystem für Sehende mit besonderen Stärken im Formelsatz, wodurch es sich auch als Mathematikschrift auf dem Computer verwenden läßt. Sehende und Blinde können den Quelltext mit jedem beliebigen Texteditor schreiben, lesen und bearbeiten. Dabei können sie für mathematische Ausdrücke den LaTeX-Code zunächst benutzen, ohne an LaTeX als Satzsystem zu denken. Blinde Schüler und ihre Lehrer können die mathematischen Ausdrücke leicht lernen und über die Computertastatur problemlos eingeben. Beide können für die tägliche Arbeit den Quelltext benutzen, ohne an die spätere Kompilierbarkeit zu denken. Strukturbefehle und mathematische Umgebungen werden weggelassen.

Als Satzsystem hat mich LaTeX zuerst nicht überzeugt. Es schien mir umständlich und antiquiert. Gerade zu diesem Zeitpunkt erhielt ich den Auftrag, für einen blinden Studenten die schriftlichen Aufgaben für die Realschullehrerprüfung im Fach Mathematik zu übertragen. Er arbeitete an einer allgemeinen Schule und erstellte alle Texte für sich und seine sehenden Schüler in LaTeX. Beim Übertragen der Prüfungsaufgaben stellte ich fest, dass dies ungeahnte Vorteile hatte. Obwohl ich manches mathematisch gar nicht verstand, konnte ich nach dem Kompilieren sofort sehen, ob ich die Aufgaben richtig übertragen hatte. Bei der Übertragung von Abituraufgaben in Marburger Systematik hatte ich immer ein mulmiges Gefühl gehabt.

Das Prüfungsamt konnte die Datei mit den Aufgaben, die der Student bekam, lesen und die Prüfer erhielten sofort nach der Prüfung die gedruckten Lösungen in Buchdruckqualität. Ein „Dolmetscher“ war nicht nötig.

Auf Grund dieser Erfahrung bin ich der Ansicht, dass es sich für Blinde, die LaTeX als

mathematischen Code benutzen, durchaus lohnt, sich auch mit LaTeX als Satzsystem zu befassen.

Man kann die Befehle relativ schnell lernen, die nötig sind, um einen perfekten Schwarzschriftausdruck zu erzeugen. Der Vorteil: Bei LaTeX behalten auch Blinde immer die Kontrolle über das Layout des Schwarzschriftdokumentes. Die Befehle zur Gestaltung des Dokumentes stehen direkt im Text. Wer ihre Bedeutung kennt, weiß wie sie sich auswirken und dass sie von LaTeX nach allen Regeln der Buchdruckerkunst umgesetzt werden.

So gesehen ist LaTeX vielleicht die ideale Textverarbeitung für Blinde, unter dem Motto: „What you say is what you get“. Man muss LaTeX eben nur sagen, was man will. Dagegen war der Slogan: „What you see is what you get“ für Blinde schon immer eine zweifelhafte Verheißung.

Der Einwand, dass die DVI-Dateien, von LaTeX nur von Eingeweihten betrachtet und ausgedruckt werden können, gilt übrigens längst nicht mehr. Aus dem gleichen Quelltext kann man auch PDF-Dateien erzeugen und es gibt heute kaum einen Windows Rechner, auf dem der Acrobat Reader nicht installiert ist. Damit kann jeder Sehende die PDF-Dateien anzeigen und drucken. Auch zu HTML gibt es Schnittstellen und Folien für Präsentationen können ebenso mit LaTeX erstellt werden.

## 2 Winword - vom Editor zur TeXShell

Winword ist heute die Standard-Textverarbeitung. Wer LaTeX nur einmal ausprobieren will, möchte seinen Text mit der gewohnten Textverarbeitung schreiben. Außerdem ist man vielleicht noch unsicher und will das Ergebnis während der Arbeit ständig kontrollieren. OK, das können Sie haben. Dieser Text wurde mit Word für Windows geschrieben und während der Arbeit wiederholt mit MikTeX kompiliert und sofort im DVI-Viewer betrachtet.

Ein Grund für die verbreitete Scheu, LaTeX zu benutzen, ist die ungewohnte Syntax der LaTeX-Befehle. Anfangs macht man leicht kleine Fehler, die sich bei der Kompilierung aber fatal auswirken können. Sehende bevorzugen deshalb Shells mit Symbolleisten, in denen man nur das gewünschte Symbol anklicken muss, damit der entsprechende LaTeX-Befehl an der Cursorposition eingefügt wird.

Eine ähnliche Arbeitstechnik wollten wir in Word auch für Blinde ermöglichen. Da uns Symbolleisten für Blinde nicht geeignet erschienen, versuchten wir die Lösung über Menübefehle. Dies erwies sich als hilfreiche Methode zum Erlernen der LaTeX-Befehle.

Wir haben Winword zur LaTeX-Shell zu erweitert. Dazu wurde der zusätzliche Menüpunkt `TeXShell` in die Menüleiste aufgenommen. Darunter findet man die Shell-Kommandos, um DVI-Dateien zu erzeugen und anzuzeigen. So kann man jetzt jederzeit während der Arbeit den momentanen Stand der aktiven Datei kompilieren

Alt-S-K (`TeXShell/kompilieren`)

Sehende können das Ergebnis sofort im DVI-Viewer betrachten.

Alt-S-D (`TeXShell/darstellen`)

Wahlweise kann man auch PDF-Dateien erzeugen.

Alt-S-P (`TeXShell/PDF erzeugen`)

Unter dem Menüpunkt `TeXShell` findet man aber auch Untermenüs mit den wichtigsten LaTeX-Befehlen. Dahinter verbergen sich eine Reihe von Makros, die es auch dem LaTeX-Anfänger ermöglichen, gut gestaltete Dokumente zu erstellen. Darunter ist der Vorspann für ein neues Dokument, eine Titelseite, das Inhaltsverzeichnis, verschiedene Absatz- und Schriftformate, manuelle Wechsel und auch einige mathematische Ausdrücke.

### 2.1 Vorbereitungen in Word

Wir speichern den Quellcode als Ansi-Text (in Word also „Nur Text“) mit der Erweiterung `tex`. Man muss allerdings dafür sorgen, dass die Endung `.tex` von Winword beim Speichern akzeptiert wird. Ansonsten wird ungefragt die Endung `.txt` zusätzlich angehängt und das widerspricht den LaTeX-Konventionen. Für Office 2000 in Windows 98 erzeugt man z.B. im Explorer im Menü: Ansicht/Ordneroptionen den Dateityp: „Latex Quellcode.“ Man gibt als Standarderweiterung `.tex` an und wählt als Inhaltstyp: „application/msword“. Danach kann man beim Speichern immer den Dateityp: „Nur Text“ wählen und die vorgeschlagene Endung `.txt` mit `.tex` überschreiben. Nachdem eine Datei schon die Endung `tex` hat, kann man sie nach der erneuten Bearbeitung problemlos speichern, ohne dass man die Erweiterung erneut eingeben muss.

Man kann das Problem aber auch getrost unserem Makro überlassen. Bei der ersten Kompilierung wird die aktuelle Datei in jedem Falle zuerst als ANSI-Text gespeichert und erhält gleichzeitig die Endung `.tex`, gleichgültig welches Format und welche Endung sie vorher hatte. Eine vorher gespeicherte Datei mit der ursprünglichen Endung (z.B. `.doc`) bleibt erhalten. Dagegen wird eine gleichnamige Datei mit der Endung `.tex` gnadenlos und ohne Sicherheitsabfrage überschrieben.

Ein Hinweis zum Schluss: Kompiliert man eine Datei, die vorher noch nicht gespeichert wurde, wird sie im Hauptverzeichnis von Laufwerk C unter dem Namen `Docum.tex` abgelegt und kompiliert. Es empfiehlt sich immer, eine neue Datei vor der Kompilierung im gewünschten Ordner zu speichern.

## 2.2 Umlaute und Sonderzeichen

Wer an Textverarbeitung mit Word gewöhnt ist, will bestimmt die Umlaute und Sonderzeichen ganz normal eingeben. Es gibt keinen Grund mehr, dies nicht zu tun. Wenn Umlaute und andere Sonderzeichen im Quelltext stehen, muss man aber darauf achten, dass im Vorspann folgende Zeilen stehen:

```
\usepackage[ansinew]{inputenc}
```

Bei Verwendung der erweiterten EC-Zeichensätze braucht man auch die Zeile

```
\usepackage{t1enc}.
```

Alle Sonderzeichen werden dann richtig interpretiert. Man kann dann auch das Gradzeichen und Ähnliches direkt von der Tastatur eingeben. Wenn man PDF-Dateien erzeugt, sollte man allerdings die EC-Zeichensätze meiden, da die Ergebnisse im Acrobat Reader sehr bescheiden aussehen.

Die Probleme mit den Umlauten und Sonderzeichen kann man natürlich auch umgehen, wenn man die nationalen Sonderzeichen nach den klassischen LaTeX-Konventionen eingibt. Ich konnte mich allerdings nie dazu durchringen, da die Lesbarkeit des Quelltextes für Ungeübte leidet. Vertreter der „reinen Lehre“ werden natürlich sagen, das sei nur Gewohnheit und gerade jetzt bei den deutschen Anführungszeichen muss ich ja auch „befremdliche“ Zeichen eingeben. Dafür haben wir vorerst keine bessere Lösung.

## 2.3 Umschließende Befehle eingeben

Befehle, die einen Text umschließen, können mit unseren Makros auf zwei Arten eingefügt werden. Entweder man erzeugt zuerst den Befehl einschließlich der notwendigen Klammern, z.B.:

```
\section{}
```

dann befindet sich der Cursor anschließend an der Stelle, wo der Text eingegeben werden muss. Wenn der Text jedoch schon vorhanden ist, kann man ihn zuerst markieren und dann den Befehl erzeugen. Der Text wird dann korrekt umschlossen. Dies entspricht der Arbeitsweise, die man von der normalen Textverarbeitung in Word – z.B. bei der nachträglichen Formatierung von Textteilen – gewohnt ist.

Beispiel: `\textbf{fette Schrift}`

## 3 Die Menübefehle unserer TeXShell mit Beispielen

### 3.1 Das Seitenlayout

Das Seitenformat für Standard-Dokumente wird mit Alt-S-N (TeXShell/Neu) erzeugt. Hinter dem Befehl:

```
\input{vorspann}
```

verbirgt sich die Definition unserer Standardseite, für die LaTeX Dokumentklasse article mit den von uns bevorzugten Seiteneinstellungen. Dieser Vorspann kann individuell gestaltet werden und muss unter dem Namen vorspann.tex innerhalb des TeX-Systems gespeichert werden. Wir haben im Ordner localtexmf den Unterordner myinput erzeugt und speichern dort alle eigenen Input-Dateien. Nachdem eine neue Datei dort abgelegt wurde, muss jeweils die Datenbank aufgefrischt werden. Das geschieht im Windows Startmenü unter: Programme - MikTeX - MikTeX Options - Refresh Now. Danach findet MikTeX die Inputs immer, gleichgültig wo die Datei liegt, welche kompiliert werden soll.

Den gesamten Inhalt des Vorspanns kann man mit dem Befehl Alt-S-V in ein neues Dokument einfügen. Diesen Vorspann kann man nach den eigenen Bedürfnissen und Wünschen anpassen und ergänzen. In diesem Fall ist der Befehl input bzw. die Datei vorspann.tex nicht notwendig.

### 3.2 Das Untermenü Einfügen (Alt-S-E)

#### 3.2.1 Titelseite (Alt-S-E-T)

Der Menüpunkt: TeXShell/Einfügen/Titelseite erzeugt im Dokument eine Titelseite, in der nur Titel, Autor und Datum eingesetzt werden müssen. Das Titelblatt für das vorliegende Dokument zeigt das Ergebnis.

#### 3.2.2 Inhaltsverzeichnis (Alt-S-E-V)

Der Menüpunkt: TeXShell/Einfügen/Inhaltsverzeichnis enthält alles, was nötig ist, damit LaTeX automatisch ein Inhaltsverzeichnis mit Seitennummerierung erstellt. Achtung! Das Inhaltsverzeichnis wird normalerweise erst nach der 2. Kompilierung eingefügt. Wir benutzen deshalb den erweiterten Miktex-Befehl texify an Stelle des Befehls latex. In TeX-live ist dieser Befehl nicht enthalten. Wer dieses System benutzt, muss das entsprechende Makro anpassen. Auch der Aufruf des Viewers muss in diesem Fall geändert werden.

#### 3.2.3 Fußnoten (Alt-S-E-F)

Fußnoten werden mit dem Menüpunkt:

TeXShell/Einfügen/Fußnote erzeugt. Dieser Absatz enthält zwei Fußnoten<sup>1</sup>. Der Fußnotentext wird an der entsprechenden Textstelle in den Text eingefügt. Das ist für den Blinden beim Lesen und Bearbeiten der richtige Platz, da er sonst das Seitenende suchen

---

<sup>1</sup>Dies ist die erste Fußnote



müsste. Beim Ausdruck wird der Fußnotentext von LaTeX aber immer am Seitenende ausgedruckt, wie es den klassischen Regeln<sup>2</sup> der Druckgestaltung entspricht.

### 3.2.4 Zeilenumbruch (Alt-S-E-Z)

Zeilenumbrüche innerhalb von Absätzen kann man durch einen doppelten Backslash erzwingen. Dies entspricht dem Shift+Return in der normalen Textverarbeitung. Als Shortcut steht bei uns auch AltGr+Return zur Verfügung.

### 3.2.5 Seitenumbruch (Alt-S-E-S)

Der erzwungene Seitenumbruch (pagebreak) entspricht der Tastenkombination Strg+Return in der normalen Textverarbeitung.

## 3.3 Überschriften (Alt-S-Ü)

Die Überschrift für Gliederungsebene 1 wurde mit dem Menüpunkt:

TeXShell/Überschrift/1. Ebene erzeugt.

Entsprechend werden die Überschriften für die Gliederungsebenen 2 und 3 mit Ü2 und Ü3 aufgerufen. LaTeX nummeriert die Gliederung automatisch und übernimmt sie ins Inhaltsverzeichnis. Man kann also auch nachträglich noch beliebig viele Überschriften einfügen oder herauslöschen, ohne sich um die Zählung kümmern zu müssen. Will man keine automatische Nummerierung, verwendet man die section-Befehle mit einem zusätzlichen \*.

```
\section*{} \subsection*{} \subsubsection*{}
```

## 3.4 Absatzformate

### 3.4.1 Fließtext

Ich schreibe einfach drauflos. Sollten übrigens im Text Silbentrennungen vorkommen, so sind sie alle von LaTeX automatisch erzeugt. Ich habe mir darüber keine Gedanken gemacht.

Dies ist jetzt ein neuer Abschnitt, ich habe ihn durch zwei Returns (also durch eine Leerzeile) erzwungen. Er reißt auch jetzt nicht ab, obwohl ich mehrere einzelne Carriage>Returns ohne Leerzeile eingefügt habe.

Erste Regel: Schreiben Sie wie sie wollen, kümmern Sie sich nicht um Zeilenumbrüche, Abstände, überflüssige Leerzeichen und Trennungen. Wenn Sie aber wirklich einen Absatz machen wollen, müssen Sie zweimal Return drücken.

---

<sup>2</sup>Dies ist die zweite Fußnote

### 3.4.2 Absatz linksbündig (Alt-S-A-L)

Der Menüpunkt: TeXShell/Absatz/links erzeugt Flattersatz linksbündig. Normalerweise wird unter LaTeX, wie im Buchsatz üblich, immer der Blocksatz verwendet. Benutzen Sie Flattersatz nur dort, wo er unbedingt nötig ist.

### 3.4.3 Absatz rechtsbündig (Alt-S-A-R)

Rechtsbündige Absätze werden mit: TeXShell/ Absatz/rechts nach dem gleichen Verfahren erzeugt.

### 3.4.4 Absatz zentriert (Alt-S-A-Z)

Der Menüpunkt: TeXShell/Absatz/zentriert erzeugt zentrierte Absätze, wie sie z.B. in Gedichten vorkommen.

### 3.4.5 Absatz eingerückt (Alt-S-A-E)

Dieser Absatz ist an beiden Seiten um einen Zentimeter eingerückt. Der Text wird zwischen dem `\begin{quote}` und dem `\end{quote}` Kommando eingefügt.

Wie Sie sehen kann man Einrückungen auch schachteln!

Dank eines Makros kann man das passende Absatzformat mit drei Tastendrücken erzeugen.

### 3.4.6 Absatz nummeriert (Alt-S-A-N)

1. Nun sollen die nächsten Absätze nummeriert werden. Die Nummerierung wird von LaTeX automatisch vorgenommen. Nummerierungen können geschachtelt werden. Die zweite Ebene wird nach der klassischen Gliederungsstruktur in die Punkte (a), (b), usw. unterteilt.
  - (a) Dies ist ein Unterpunkt zu dieser Aufzählung. Die Einrückung im Quelltext ist nur zur klareren Übersicht beim Editieren gemacht, eigentlich aber unnötig.
  - (b) Dies ist ein zweiter Unterpunkt zu dieser Aufzählung.
3. Das ist Absatz Nummer drei.
4. Dies ist Absatz Nummer vier.

### 3.4.7 Absatz Spiegelstrich (Alt-S-A-S)

- Nun kommen Spiegelstrichabsätze, (hier mit einem vergrößerten Punkt versehen). Auch Spiegelstriche kann man schachteln. Die zweite Ebene wird dann mit - (Gedankenstrich) eingeleitet. Die dritte Ebene mit einem zentrierten Stern (\*).
- Das ist der zweite Spiegelstrichabsatz, eingeleitet mit dem Befehl `item`, den ich mit dem Kürzel `it` aufrufe
- Das ist der dritte Spiegelstrichabsatz.
  - Das ist jetzt ein Unterpunkt zu Spiegelstrich Nummer drei.
    - \* hier beginnt die dritte Ebene.
    - \* Das ist noch ein Punkt der dritten Ebene.
    - \* Dritter Punkt der Ebene drei.
  - Das ist der zweite Unterpunkt zu Spiegelstrich drei.
  - Das ist der dritte Unterpunkt zu Spiegelstrich drei.
- Das ist der vierte Spiegelstrich, diesmal etwas länger, damit er über mehrere Zeilen geht. Er soll sogar aus zwei abgesetzten Absätzen bestehen.

Das ist jetzt der zweite Unterabsatz im selben Spiegelstrich. Das geht also auch.

### 3.5 Tabellen (Alt-S-A-T)

Bei Tabellen wird die Struktur der Tabelle in der ersten Zeile beschrieben (einschließlich der vertikalen Linien). Für das nachfolgende Beispiel steht in der ersten Zeile:

```
\begin{tabular}{|r||l|c|c|}
```

`l` für linksbündig, `c` für zentriert, `r` für rechtsbündig, `|` für vertikale Linie

In die folgenden Zeilen werden die Tabelleneinträge eingefügt. Als Tabulatorzeichen dient das kaufmännische Undzeichen. Der Zeilenumbruch wird durch doppelten Backslash gekennzeichnet. Horizontale Linien werden durch das Kommando `\hline` am Ende der Zeile erzeugt. Im nachfolgenden Beispiel ist die Tabelle zusätzlich zentriert.

Platz	Verein	Tore	Punkte
1.	Bay. München	66:33	51
2.	Hamburger SV	65:34	50
3.	Werder Bremen	52:43	46
4.	...	...	...

Da man die Spalten von Tabellen von Fall zu Fall verschieden gestalten muss, stellt das Makro: Alt-S-A-T: nur die Grundstruktur zur Verfügung. Die Anzahl der Spalten und ihre Ausrichtung muss man jeweils selbst festlegen.

### 3.5.1 Tabellen mit mehrzeiligen Spalten

Will man Tabellen erstellen, die nicht nur einzeilige Einträge zulassen, kann man die Spalten auch als Parbox vorgeben. Im nachfolgenden Falle sind 2 Spalten definiert. Die erste ist 4 cm breit, die zweite 12 cm.

```
\begin{tabular}{|p{4cm}| p{12cm}|}
```

Spalte 1	Spalte 2
4 cm breite Spalte	12 cm breite Spalte
Stichwort	Hier ist der Text auf der rechten Seite etwas länger, damit er nicht in eine Zeile passt. Der Inhalt einer jeden Parbox steht in geschweiften Klammern. Wie in einzeiligen Tabellen, können auch senkrechte und waagrechte Linie zur Abgrenzung der Parboxen benutzt werden
Jetzt ist der Text der linken Spalte länger	Stichwort

### 3.5.2 Kombinationen von Tabellenstrukturen

Selbst die Kombination beider Strukturierungsmöglichkeiten ist erlaubt.

```
\begin{tabular}{r p{15cm}}
```

- 1 Das ist eine Tabelle, in der eine rechtsbündige Spalte mit einer Parbox von 15 cm kombiniert ist. Der Text in der Parbox ist länger als eine Zeile.

1000 Aber Vorsicht! Das funktioniert nur richtig, wenn gewährleistet ist, dass die Breite der Tabelle nicht die Seitenbreite überschreitet.

## 3.6 Schriftarten, Schriftattribute und Schriftgrößen

### 3.6.1 Schriftarten

Die normale Schriftart in LaTeX ist Roman. Wir nennen sie der Einfachheit halber Times, da die meisten Menschen diesen Schrifttyp unter dieser Bezeichnung kennen. Sans Serif nennen wir aus dem gleichen Grund Helvetica und für Teletype verwenden wir die Bezeichnung Maschinenschrift. Man kann jederzeit in eine andere Schriftart wechseln, indem man das Kommando der neuen Schriftart setzt.

Alt-S-Z-T Schrift Times  
Alt-S-Z-H Schrift Helvetica  
Alt-S-Z-M Maschinenschrift

### 3.6.2 Schriftattribute

Um einen Textteil **fett** zu formatieren, wird einfach der Menüpunkt Alt-S-S-F (fett) aufgerufen. Hier gelten die Regeln für die Eingabe umschließender Befehle, ebenso wie für die anderen Schriftattribute. Die entsprechenden Menüpunkte sind:

Alt-S-Z-F **fett**  
Alt-S-Z-G *geneigt*  
Alt-S-Z-K *kursiv*  
Alt-S-Z-C CAPITALS (SMALL CAPS)  
Alt-S-Z-U unterstrichen

Unterstreichungen sind im Buchdruck verpönt. Wir wollten diesen Menüpunkt zuerst ganz weglassen. Es geht aber mit dem Befehl:

`\underline{}`.

### 3.6.3 Schriftgrößen

Für Schriftgrößen haben wir das Untermenü Alt-S-S-G. Die entsprechenden LaTeX-Befehle kann man sich leicht merken:

Tastenfolge	Größe und Aussehen	Befehl
Alt-S-Z-G-1	Größe 1 (ca. 1.2 mm):	<code>\tiny</code>
Alt-S-Z-G-2	Größe 2 (2 mm):	<code>\scriptsize</code>
Alt-S-Z-G-3	Größe 3 (2,5 mm):	<code>\footnotesize</code>
Alt-S-Z-G-4	Größe 4 (3 mm):	<code>\small</code>
Alt-S-Z-G-5	Größe 5 (3,5 mm):	<code>\normalsize</code>
Alt-S-Z-G-6	Größe 6 (4 mm):	<code>\large</code>
Alt-S-Z-G-7	Größe 7 (4,5 mm):	<code>\Large</code>
Alt-S-Z-G-8	Größe 8 (5 mm):	<code>\LARGE</code>
Alt-S-Z-G-9	Größe 9 (6 mm):	<code>\huge</code>
Alt-S-Z-G-0	Größe 10 (6,5 mm):	<code>\Huge</code>

## 3.7 Mathematik

Mathematische Ausdrücke werden in LaTeX nur dann richtig interpretiert und dargestellt, wenn sie innerhalb einer mathematischen Umgebung auftreten. Man muss LaTeX in den mathematischen Modus versetzen und danach wieder in den Textmodus umschalten. Bestimmte Befehle sind nur im mathematischen Modus zulässig und führen im Textmodus zu Fehlermeldungen.

### 3.7.1 Mathetext oder abgesetzte Formeln (Alt-S-M-X oder Alt-S-M-F)

Der nachfolgende Text enthält mathematische Ausdrücke, wie z. B. den folgenden:  $\frac{2x+5}{2y-5} = \sqrt[3]{2\delta_i}$ . Diesen Ausdruck ist hier zwischen zwei Dollarzeichen eingeschlossen, um die mathematische Umgebung zu kennzeichnen. Der mathematische Ausdruck wird in diesem Falle fortlaufend in den Text eingefügt, ohne Rücksicht auf den Zeilenumbruch. Statt des Dollarzeichens hätte man auch Backslash und runde Klammern über die Tastatur eingeben können.

Wenn man die Formel schön abgesetzt in einer Zeile haben will, verwendet man eine andere mathematische Umgebung, die ich mit dem Menüpunkt TeXShell/Mathe/Formel erzeugt wird.

Das sieht dann so aus:

$$\frac{2x + 5}{2y - 5} = \sqrt[3]{2\delta_i}$$

Standardmäßig sind abgesetzte Formeln zentriert. In diesem Dokument sind sie ausnahmsweise linksbündig, weil im Vorspann die Option fleqn in der ersten Zeile (documentclass) eingefügt wurde. Normalerweise werden die Formeln dabei um 1 cm eingerückt. Durch den Befehl mathindent0pt weiter unten ist die Einrückung auf Null gesetzt.

Im Untermenü TeXShell/Mathe finden sich folgende Menüpunkte:

Alt-S-M-B	Bruch
Alt-S-M-F	Formel abgesetzt
Alt-S-M-X	Formel im Fließtext
Alt-S-M-G	Gleichungsfeld
Alt-S-M-I	Integral
Alt-S-M-S	Summe
Alt-S-M-P	Produkt
Alt-S-M-H	Hochzahl (Exponent)
Alt-S-M-T	Tiefzahl (Index)
Alt-S-M-W	Wurzel

### 3.7.2 Gleichungsfelder (Alt-S-M-G)

Will man z.B. die Auflösung einer Gleichung über mehrere Zeilen verteilt darstellen, verwendet man die Umgebung

```
\begin{eqnarray*}
```

`\end{eqnarray*}`

$$\begin{aligned}\frac{4x+1}{3x} - 1 &= \frac{4x+1}{3x} - \frac{3x}{3x} \\ &= \frac{4x+1-3x}{3x} \\ &= \frac{x+1}{3x}\end{aligned}$$

Damit die Gleichheitszeichen schön untereinander stehen, wurde hier das kaufmännische Undzeichen als Tabulatorzeichen vor und nach dem Gleichheitszeichen verwendet.

Sollen die Gleichungen im ganzen Dokument oder in einem bestimmten Bereich durchnummeriert werden, benutzt man die gleiche Umgebung ohne den Stern.

`\begin{eqnarray}`

`\end{eqnarray}`

das sieht dann so aus:

$$(x+1)^2 - (x-1)^2 = 5x - 4 \tag{1}$$

$$x^2 + 2x + 1 - (x^2 - 2x + 1) = 5x - 4 \tag{2}$$

$$x^2 + 2x + 1 - x^2 + 2x - 1 = 5x - 4 \tag{3}$$

$$4x = 5x - 4 \tag{4}$$

$$-x = -4 \tag{5}$$

$$x = 4 \tag{6}$$

Hier als Beispiel noch eine Horrorformel. Wer weiß, was das bedeutet?

$$\frac{4x+1}{3x} - 16\sqrt[3]{\frac{2a+4}{x^3-2y_1}} = \frac{4x\sqrt[3]{y_i^2x^2}}{\sqrt[5]{\frac{x^3y^{-4}+7(3x-y)\frac{3y}{3z}}{4(x^2\sqrt[3]{y_ix^{x-7}+3y)}}$$

Das ist absoluter Quatsch! Aber es sieht doch toll aus.

### 3.7.3 Zeilenumbrüche und Kommentare im Quelltext

In Latex kann man hinter einem Prozentzeichen Kommentare einfügen, die bei der Kompilierung ignoriert werden. Z.B.

`\sqrt{23+5}` \$ %das ist die Wurzel aus 28

Diese Möglichkeit bietet sich für den Lehrer beim Korrigieren an.

## 3.8 Übersicht der Befehle im Menü TeXShell in der Vorlagen-Datei latex.dot

### 3.8.1 Dokumente erstellen

Tastenfolge	Verwendungszweck
Alt-S-N	Neues Dokument (article)
Alt-S-K	Aktives Dokument kompilieren
Alt-S-D	Aktives Dokument im DVI-Viewer darstellen

### 3.8.2 Überschriften

Tastenfolge	Verwendungszweck	Befehl
Alt-S-Ü-1	Überschrift Ebene 1	<code>\section{ }</code>
Alt-S-Ü-2	Überschrift Ebene 2	<code>\subsection{ }</code>
Alt-S-Ü-3	Überschrift Ebene 3	<code>\subsubsection{ }</code>

### 3.8.3 Absatzformate

Tastenfolge	Verwendungszweck	Befehl
Alt-S-A-E	Absatz eingerückt	<code>\begin{quote}</code> <code>\end{quote}</code>
Alt-S-A-L	Absatz linksbündig	<code>\begin{flushleft}</code> <code>\end{flushleft}</code>
Alt-S-A-N	Absatz nummeriert	<code>\begin{enumerate}</code> <code>\end{enumerate}</code>
Alt-S-A-R	Absatz rechtsbündig	<code>\begin{flushright}</code> <code>\end{flushright}</code>
Alt-S-A-S	Absatz Spiegelstrich	<code>\begin{itemize}</code> <code>\end{itemize}</code>
Alt-S-A-T	Absatz Tabelle	<code>\begin{tabular}{}{}</code> <code>\end{tabular}</code>
Alt-S-A-Z	Absatz zentriert	<code>\begin{center}</code> <code>\end{center}</code>



### 3.8.4 Allgemeine Elemente im Dokument

Tastenfolge	Verwendungszweck	Befehl
Alt-S-E-D	aktuelles Datum	<code>\today</code>
Alt-S-E-F	Fußnote	<code>\footnote</code>
Alt-S-E-I	Item (Beginn von Aufzählungspunkten)	<code>\item</code>
Alt-S-E-S	Seitenumbruch	<code>\pagebreak</code>
Alt-S-E-Z	Zeilenumbruch	<code>\\</code>
Alt-S-E-V	Inhaltsverzeichnis	<code>\tableofcontents</code>
Alt-S-E-T	Titelseite	<code>\begin{titlepage}</code>
	Name des Autors	<code>\author{}</code>
	Titel (auch mehrzeilig)	<code>\title{}</code>
	Erstelldatum (kann auch fest eingetippt werden)	<code>\date{\today}</code>
		<code>\end{titlepage}</code>
		<code>\maketitle</code>

### 3.8.5 Mathematische Befehle

Tastenfolge	Verwendungszweck	Befehl
Alt-S-M-B	Mathe Bruch	<code>\frac{ }{ }</code>
Alt-S-M-F	Mathe Formel abgesetzt	<code>\[ \]</code>
Alt-S-M-H	Hochzahl (Exponent)	<code>^{ }</code>
Alt-S-M-I	Mathe Integral	<code>\int_{ }^{ }</code>
Alt-S-M-S	Mathe Summe	<code>\sum_{ }^{ }</code>
Alt-S-M-P	Mathe Produkt	<code>\prod_{ }^{ }</code>
Alt-S-M-X	Mathe Formel im Text	<code>\$ \$</code>
Alt-S-M-T	Tiefzahl (Index)	<code>_{ }</code>
Alt-S-M-W	Mathe Wurzel	<code>\sqrt[ ]{ }</code>
Alt-S-M-G	Mathe Gleichungsfeld	<code>\begin{eqnarray*}</code> <code>\end{eqnarray*}</code>

### 3.8.6 Schriftarten

Tastensequenz	Verwendungszweck	Befehl
Alt-S-S-T	Schrift normal (TimesNewRoman)	<code>\textrm{ }</code>
Alt-S-S-H	Helvetica (Arial)	<code>\textsf{ }</code>
Alt-S-S-M	Maschinenschrift (Courier)	<code>\texttt{ }</code>

### 3.8.7 Schriftattribute

Tastensequenz	Verwendungszweck	Befehl
Alt-S-S-C	SCHRIFT CAPITALS (SMALL CAPS)	<code>\textsc{ }</code>
Alt-S-S-F	<b>Schrift fett (bold)</b>	<code>\textbf{ }</code>
Alt-S-S-G	<i>Schrift geneigt (slanted)</i>	<code>\textsl{ }</code>
Alt-S-S-K	<i>Schrift kursiv (italic)</i>	<code>\textit{ }</code>

### 3.8.8 Schriftgrößen

Tastensequenz	Größe und Aussehen	Befehl
Alt-S-S-G-1	Größe 1 (ca. 1,2 mm):	<code>\tiny</code>
Alt-S-S-G-2	Größe 2 (2 mm):	<code>\scriptsize</code>
Alt-S-S-G-3	Größe 3 (2,5 mm):	<code>\footnotesize</code>
Alt-S-S-G-4	Größe 4 (3 mm):	<code>\small</code>
Alt-S-S-G-5	Größe 5 (3,5 mm):	<code>\normalsize</code>
Alt-S-S-G-6	Größe 6 (4 mm):	<code>\large</code>
Alt-S-S-G-7	Größe 7 (4,5 mm):	<code>\Large</code>
Alt-S-S-G-8	Größe 8 (5 mm):	<code>\LARGE</code>
Alt-S-S-G-9	Größe 9 (6 mm):	<code>\huge</code>
Alt-S-S-G-0	Größe 10 (6,5 mm):	<code>\Huge</code>

## 4 Das griechische Alphabet

Die Regel ist ganz einfach. Verwende den Namen des Buchstabens als LaTeX-Befehl.

Beispiel:  $\alpha$  erzeugt man mit `\alpha`

Mein Problem sind die Großbuchstaben. Alle griechischen Großbuchstaben, die genauso aussehen wie die entsprechenden lateinischen Buchstaben, dürfen nicht nach dieser Regel behandelt werden.

In all diesen Fällen darf man nur den lateinischen Großbuchstaben verwenden. Nur dort, wo ein eigener griechischer Großbuchstabe existiert, muss man den Backslash mit dem Buchstabenwort verwenden, der erste Buchstabe wird dabei großgeschrieben.

### 4.0.9 Zwei verschiedene Zeichen gibt es bei folgenden Buchstaben

$\gamma$	$\Gamma$	<code>\gamma</code>	<code>\Gamma</code>
$\delta$	$\Delta$	<code>\delta</code>	<code>\Delta</code>
$\theta \vartheta$	$\Theta$	<code>\theta</code> <code>\vartheta</code>	<code>\Theta</code>
$\lambda$	$\Lambda$	<code>\lambda</code>	<code>\Lambda</code>
$\xi$	$\Xi$	<code>\xi</code>	<code>\Xi</code>
$\pi$	$\Pi$	<code>\pi</code>	<code>\Pi</code>
$\sigma \varsigma$	$\Sigma$	<code>\sigma</code> <code>\varsigma</code>	<code>\Sigma</code>
$\upsilon$	$\Upsilon$	<code>\upsilon</code>	<code>\Upsilon</code>
$\phi \varphi$	$\Phi$	<code>\phi</code> <code>\varphi</code>	<code>\Phi</code>
$\psi$	$\Psi$	<code>\psi</code>	<code>\Psi</code>
$\omega$	$\Omega$	<code>\omega</code>	<code>\Omega</code>

### 4.0.10 Bei folgenden Buchstaben gibt es keinen eigenen Großbuchstaben

$\alpha$	A	<code>\alpha</code>	A (lateinisch)
$\beta$	B	<code>\beta</code>	B
$\epsilon$	E	<code>\epsilon</code>	E
$\zeta$	Z	<code>\zeta</code>	Z
$\eta$	H	<code>\eta</code>	H
$\iota$	I	<code>\iota</code>	I
$\kappa$	K	<code>\kappa</code>	K
$\mu$	M	<code>\mu</code>	M
$\nu$	N	<code>\nu</code>	N
$\rho \varrho$	P	<code>\rho</code> <code>\varrho</code>	P
$\tau$	T	<code>\tau</code>	T
$\chi$	X	<code>\chi</code>	X

o und O sind auch im griechischen Alphabet enthalten, dürfen aber nur als lateinische Buchstaben eingegeben werden.