

L^AT_EX– Publizieren für Spezialisten

Xenia Rendtel

29. Mai 2004

Einleitung

Die Geschichte

Ziel des Kurses

Die Installation

Grundlagen

Literatur

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.
- ▶ „What you see ist what you get“

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.
- ▶ „What you see ist what you get“
- ▶ Beispiele: MS Word, Framemaker

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.
- ▶ „What you see ist what you get“
- ▶ Beispiele: MS Word, Framemaker

2. Markup-Systeme (Textformatierungsprogramme):

- ▶ Autor schreibt das mit speziellen Befehlen versehene Dokument in eine Datei.

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.
- ▶ „What you see ist what you get“
- ▶ Beispiele: MS Word, Framemaker

2. Markeup-Systeme (Textformatierungsprogramme):

- ▶ Autor schreibt das mit speziellen Befehlen versehene Dokument in eine Datei.
- ▶ Befehle legen die logische Struktur und das Layout fest.

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.
- ▶ „What you see ist what you get“
- ▶ Beispiele: MS Word, Framemaker

2. Markup-Systeme (Textformatierungsprogramme):

- ▶ Autor schreibt das mit speziellen Befehlen versehene Dokument in eine Datei.
- ▶ Befehle legen die logische Struktur und das Layout fest.
- ▶ Markup-System wird auf die Datei angewendet und erzeugt weitere Datei, die das auf dem Bildschirm ansehbare oder druckbare endgültige Dokument enthält.

Ansätze zur Textformatierung

1. WYSIWYG-Systeme:

- ▶ Autor legt das Layout eines Dokumentes bei der interaktiven Eingabe fest.
- ▶ „What you see ist what you get“
- ▶ Beispiele: MS Word, Framemaker

2. Markup-Systeme (Textformatierungsprogramme):

- ▶ Autor schreibt das mit speziellen Befehlen versehene Dokument in eine Datei.
- ▶ Befehle legen die logische Struktur und das Layout fest.
- ▶ Markup-System wird auf die Datei angewendet und erzeugt weitere Datei, die das auf dem Bildschirm ansehbare oder druckbare endgültige Dokument enthält.
- ▶ Beispiele: T_EX, L^AT_EX, HTML

Beispiel für Markeup (\LaTeX)

$$\mathbf{M} = \begin{pmatrix} \frac{1}{\sqrt{x^2+y^2}} & 0 & 0 \\ 0 & \sqrt{\sqrt{y^2 + 3z^3} + x + 1} & 0 \\ 0 & 0 & x \frac{1}{y^2+z^2} \end{pmatrix}$$

Die entsprechende \LaTeX -Datei

```
\documentclass[12pt]{article}

\begin{document}

\begin{eqnarray*}
\mathbf{M}
&= &
\left(
\begin{array}{ccc}
\frac{1}{\sqrt{x^2+y^2}} & 0 & 0 \\
0 & \sqrt{\sqrt{y^2 + 3z^3}} + x + 1 & 0 \\
0 & 0 & x^{\frac{1}{y^2 + z^2}}
\end{array}
\right)
\end{eqnarray*}

\end{document}
```

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente
- ▶ kompakte Eingabedateien

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente
- ▶ kompakte Eingabedateien
- ▶ frei verfügbar

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente
- ▶ kompakte Eingabedateien
- ▶ frei verfügbar
- ▶ Formatierung unabhängig von Ausgabegerät

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente
- ▶ kompakte Eingabedateien
- ▶ frei verfügbar
- ▶ Formatierung unabhängig von Ausgabegerät
- ▶ viele gute Bücher

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente
- ▶ kompakte Eingabedateien
- ▶ frei verfügbar
- ▶ Formatierung unabhängig von Ausgabegerät
- ▶ viele gute Bücher
- ▶ `news:de.comp.text.tex`, `news:comp.text.tex`

Die Vorteile von \LaTeX

- ▶ Viele Präsentationsvorlagen liegen bereits in \LaTeX vor
- ▶ guter mathematischer Formelsatz
- ▶ portables Dateiformat (Windows, Linux, ...)
- ▶ Unterstützung großer Dokumente
- ▶ Unterstützung mehrsprachiger Dokumente
- ▶ kompakte Eingabedateien
- ▶ frei verfügbar
- ▶ Formatierung unabhängig von Ausgabegerät
- ▶ viele gute Bücher
- ▶ `news:de.comp.text.tex`, `news:comp.text.tex`
- ▶ ganz viele Automatismen

Die Nachteile von \LaTeX

- ▶ Bedienung

Die Nachteile von \LaTeX

- ▶ Bedienung
- ▶ Editieren-Kompilier-Zyklus

Die Nachteile von \LaTeX

- ▶ Bedienung
- ▶ Editieren-Kompilier-Zyklus
- ▶ keine Menüs: kein Suchen und Ausprobieren

Die Nachteile von \LaTeX

- ▶ Bedienung
- ▶ Editieren-Kompilier-Zyklus
- ▶ keine Menüs: kein Suchen und Ausprobieren
- ▶ Änderungen am Layout nicht ganz trivial

Die Nachteile von \LaTeX

- ▶ Bedienung
- ▶ Editieren-Kompilier-Zyklus
- ▶ keine Menüs: kein Suchen und Ausprobieren
- ▶ Änderungen am Layout nicht ganz trivial
- ▶ kein professioneller Support

Geschichtliche Grundlagen



Johannes Gutenberg
(1399-1468)

Der Buchdruck

vor Gutenbergs Zeiten:

- ▶ Holzdruck
- ▶ aufwendiges und langwieriges Verfahren

Der Buchdruck

vor Gutenbergs Zeiten:

- ▶ Holzdruck
- ▶ aufwendiges und langwieriges Verfahren

Gutenbergs Idee:

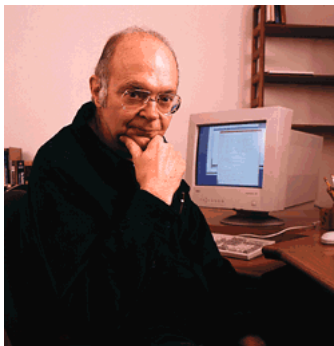
- ▶ Zerlegung des Textes in alle Einzelelemente
- ▶ Metallguss der „Lettern“
- ▶ Komposition zu Wörtern, Zeilen und Seiten

Gutenbergs Erfindung



Druckerpresse
(um 1450)

Textsatz heute (i)



Donald E. Knuth
10.01.1938 -
Erfinder von T_EX

Was ist T_EX?

1977 Donald E. Knuth

- ▶ enttäuscht über den schlechten Satz seines Buches „The Art of Computer Programming“
- ▶ Textsatzsystem (besonders für den Mathematiksatz)
- ▶ kleines Zwischenprojekt; dauerte dann viele Jahre
- ▶ eingefroren 1985, Versionsnummer ist 3.141592654

Textsatz heute (ii)



Leslie Lamport an der CAU Kiel 2003 bei Verleihung der
Ehrendoktorwürde
Entwickler von \LaTeX

Was ist \LaTeX 2.09?

1984 Leslie Lamport

- ▶ Erweiterung von \TeX damit es benutzerfreundlich wird
- ▶ Makropaket zum Setzen von Dokumenten
- ▶ baut auf \TeX auf
- ▶ Vereinfachung von komplexen \TeX -Konstrukten
- ▶ Rückgriff auf vorgefertigte Layoutelemente

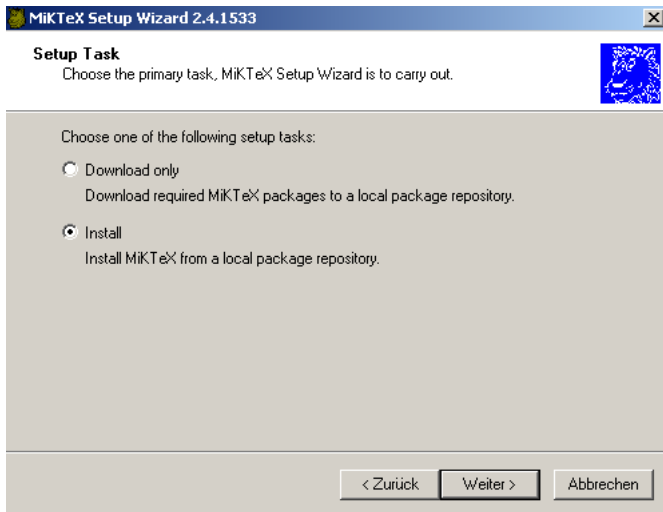
Was ist \LaTeX 2 ϵ ?

- ▶ aktuelle Version von \LaTeX
- ▶ 1994 von Leslie Lamport und anderen fertiggestellt
- ▶ erlaubt:
 - ▶ Behandlung von **Schrifttypen**
 - ▶ Verwendung von **Farben**
 - ▶ Einbindung von **Grafiken**
 - ▶ viele **neue Funktionen und Pakete**

Ziel des Kurses

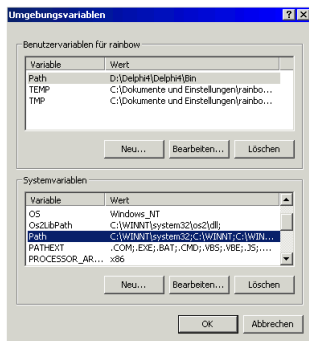
- ▶ Eigene Dokumente in \LaTeX erstellen können.

MikTeX



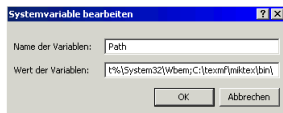
Setzen der Umgebungsvariable PATH I

- ▶ Klicken Sie rechts auf Arbeitsplatz → Eigenschaften.
- ▶ Wählen Sie die Karteikarte **Erweitert!**
- ▶ Klicken Sie auf **Umgebungsvariablen!**
- ▶ Wählen Sie aus der Liste der **Systemvariablen** die Variable **PATH** durch einmaliges Klicken aus und klicken Sie auf **Bearbeiten!**



Setzen der Umgebungsvariable PATH II

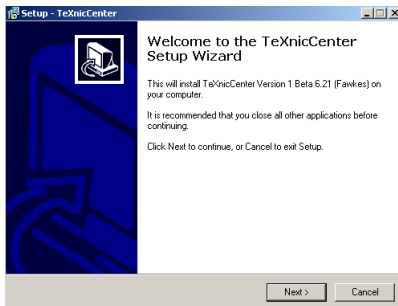
Ergänzen Sie im Feld Wert der Variablen am Zeilenende folgendes:
`;C:\texmf\miktex\bin`



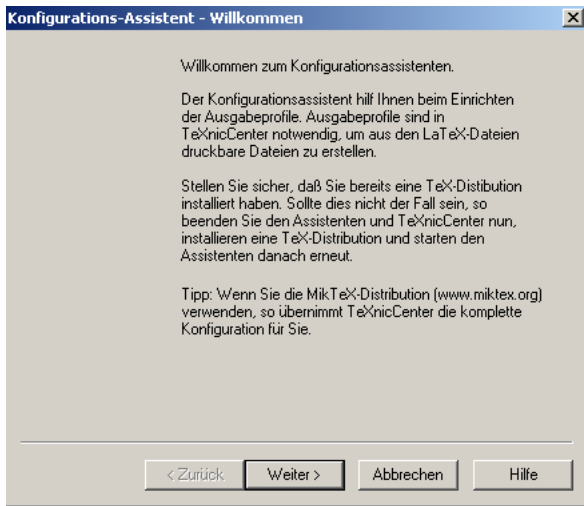
Der Editor

- ▶ Es gibt eine Vielzahl von Editoren, freie, wie auch kostenpflichtige, die sich darin unterscheiden, wie viel Arbeit sie dem Benutzer abnehmen.
- ▶ Wir nutzen das TeXnicCenter, welchen man unter <http://www.texniccenter.org/> herunterladen kann.

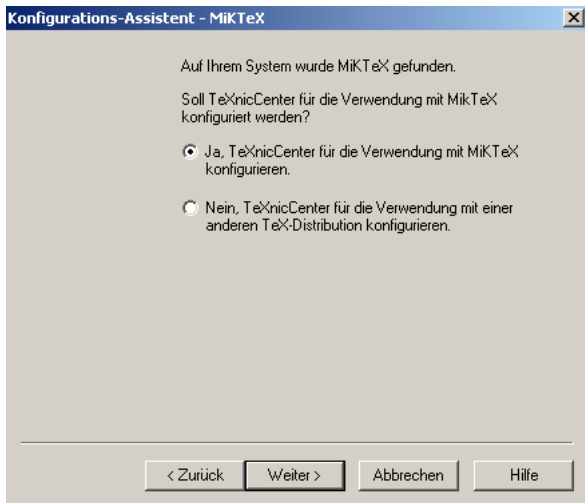
Und nun die Installation des Editors...



Die Konfiguration



Die Konfiguration



Wie funktioniert L^AT_EX?

L^AT_EX benötigt folgenden Textdateien:

- ▶ Quelldatei (`.tex`)
- ▶ Definitionsdateien (`.cls`, `.sty`) für Struktur und Layout
- ▶ Angaben zu Fontmetriken (`.tfm`)

L^AT_EX erzeugt folgende Dateien:

- ▶ geräteunabhängige Ausgabedatei (`.dvi`, `.pdf`)
- ▶ Protokoll (`.log`)
- ▶ Hilfsdateien (`.aux`, `.toc`, ...)

Allgemeine Form einer \LaTeX Datei

Schematischer Aufbau:

```
\documentclass[...Optionen...]{Klasse}
  Benutzung von Zusatzpaketen
  Einstellung von Parametern wie Texthöhe usw.
  Definition von eigenen Befehlen
\begin{document}
  Text mit Befehlen versehen
\end{document}
```

Dabei ist bei der Eingabe zu beachten:

- ▶ Absätze sind immer gut...

```
\documentclass[12pt]{article}
\begin{document}\begin{eqnarray*}
\mathbf{M} & = & \left( \begin{array}{ccc}
\frac{1}{\sqrt{x^2+y^2}} & 0 & 0 \\
0 & \sqrt{\sqrt{y^2 + 3z^3} + x + 1} & 0 \\
0 & 0 & x^{\frac{1}{y^2 + z^2}} \end{array} \right) \\
\end{eqnarray*}\end{document}
```

ist das gleiche wie:

```
\documentclass[12pt]{article}

\begin{document}

\begin{eqnarray*}
\mathbf{M}
&= & & \\
\left(
&\begin{array}{ccc}
&\frac{1}{\sqrt{x^2+y^2}} & 0 & 0 \\
0 & \sqrt{\sqrt{y^2 + 3z^3} + x + 1} & 0 \\
0 & 0 & x^{\frac{1}{y^2 + z^2}}
\end{array}
&\right)
\end{eqnarray*}

\end{document}
```

Dabei ist bei der Eingabe zu beachten:

- ▶ Eine Leerzeile entspricht einem Absatzbeginn
- ▶ Besonderheiten beachten bei Umlauten, Akzenten und Sonderzeichen wie `\`, `{`, `}` usw.:
 - ▶ Umlaute tippt man mit z.B. "a für ein ä
 - ▶ Dies kann mit speziellen Paket vermieden werden

L^AT_EX Befehle

- ▶ Ein Befehl startet immer mit einem `\`.
- ▶ Einem Befehl wird ein Argument übergeben.

```
Dies ist ein \textbf{fetter} Text.
```

Dies sieht dann wie folgt aus:

```
Dies ist ein fetter Text.
```

- ▶ Allgemeine Form eines Befehls:

```
\Befehlsname [Optional] {Zwingend}
```

Bei Befehlen zu beachten

- ▶ Bei den Befehlen wird Groß- und Kleinschreibung unterschieden
- ▶ Man kann Befehle schachteln, man muss nur auf die korrekte Klammerung achten:

```
Dies ist ein \huge{großer und \textbf{fetter} Text}
```

Dies sieht dann wie folgt aus:

```
Dies ist ein großer und fetter Text
```

Argumente des `\documentclass` -Befehls

- ▶ Einige Optionen:
 - ▶ `12pt` für 12 Punkte hohe Schrift
 - ▶ `twoside` für doppelseitigen Druck
 - ▶ `a4paper` für A4-Format
- ▶ Einige Klassen:
 - ▶ `article` für kürzere Arbeiten
 - ▶ `report` für längere Arbeiten
 - ▶ `letter` für Briefe
- ▶ Einige Zusatzpakete:
 - ▶ `\usepackage{german}` für die Anpassung an Texte in deutscher Sprache
 - ▶ `\usepackage[latin1]{inputenc}` damit man z.B. anstatt "s ß tippen kann
 - ▶ `\usepackage{amssymb}` für spezielle mathematische Symbole wie z.B. \mathbb{N}

L^AT_EX -Umgebungen

- ▶ Umgebungen beginnen mit `\begin{umgebung}` und enden mit `\end{umgebung}`
- ▶ Eine Umgebung heißt in L^AT_EX auch *Environment*
- ▶ Eine Umgebung ist z.B. eine Aufzählung, ein Zitat.

```
\begin{center}
  Dieser Text ist zentriert.
\end{center}
```

Dieser Text ist zentriert.

Das erste Dokument

```
\documentclass[12pt,twoside,a4paper]{article}

\usepackage{german}
\usepackage[latin1]{inputenc}
\usepackage{amssymb}

\begin{document}

  Das erste Grundgerüst.

\end{document}
```

Schriftgrößen

Befehl

Was passiert?

`\tiny`

Winzig

`\scriptsize`

Sehr Klein

`\footnotesize`

Fußnote

`\small`

Klein

`\normalsize`

Normal

`\large`

Groß

`\Large`

Größer

`\LARGE`

Noch Größer

`\huge`

Riesig

`\Huge`

Gigantisch

Schriftarten

Befehl	Was passiert?
<code>\textrm{text}</code>	Standardschrift
<code>\textsf{text}</code>	Schrift ohne Serifen
<code>\texttt{text}</code>	Schreibmaschinenschrift
<code>\textbf{text}</code>	Fettschrift
<code>\textit{text}</code>	<i>Kursivschrift</i>
<code>\textsl{text}</code>	<i>Schrägschrift</i>
<code>\textsc{text}</code>	KAPITÄLCHEN

Beispiel

Eine `\textit{Gruppe}` ist ein Paar...

Eine *Gruppe* ist ein Paar...

Horizontale Abstände I

Hier ist 2 cm Abstand.

Hier `\hspace{2cm}` ist 2 cm Abstand.

links

rechts

links `\hfill` rechts

Horizontale Abstände II

Befehl	Was passiert?
<code>\,</code>	sehr kleiner Abstand
<code>\enspace</code>	so breit wie eine Ziffer
<code>\quad</code>	so breit, wie ein Buchstabe hoch ist
<code>\qquad</code>	doppelt so breit wie <code>\quad</code>
<code>\hfill</code>	ein Abstand, der sich von 0 bis ∞ ausdehnen kann.

Vertikale Abstände

Hier

ist 1 cm Abstand.

Hier

```
\vspace{1cm}
```

 ist 1 cm Abstand.

Befehl

Was passiert?

```
\smallskip
```

etwa $\frac{1}{4}$ Zeile

```
\medskip
```

etwa $\frac{1}{2}$ Zeile

```
\bigskip
```

etwa 1 Zeile

```
\vfill
```

Abstand, der sich von 0 bis ∞
ausdehnen kann.

Logische Gliederung eines Dokuments

- ▶ `\part{Teil}` Nur bei `\book` und `\report`
- ▶ `\chapter{Kapitel}` Nur bei `\book` und `\report`
- ▶ `\section{Abschnitt}`
- ▶ `\subsection{Unterabschnitt}`
- ▶ `\subsubsection{Unterunterabschnitt}`

Beispiel

1 Abschnitt

1.1 Unterabschnitt

1.1.1 Unterunterabschnitt

1.1.2 Unterunterabschnitt

1.2 Unterabschnitt

2 Abschnitt

2.1 Unterabschnitt

Ausrichtung von Textteilen

Zentrierung von Textteilen durch die `center` -Umgebung:

```
\begin{center}
  Erste zu zentrierende Zeile. \\
  Zweite zu zentrierende Zeile. \\
  Dritte zu zentrierende Zeile.
\end{center}
```

Resultat:

```
Erste zu zentrierende Zeile.
Zweite zu zentrierende Zeile.
Dritte zu zentrierende Zeile.
```

Analog:

- ▶ Rechtsbündige Ausrichtung von Textteilen mittels der `flushright` -Umgebung
- ▶ Linksbündige Ausrichtung von Textteilen mittels der `flushleft` -Umgebung

Aufzählungen I

Erzeugung von Aufzählungen durch `enumerate` -Umgebung:

```
\begin{enumerate}
\item Erster Teil der Aufzählung.
  Er darf ruhig länger als eine
  Zeile sein.
\item Zweiter Teil der Aufzählung.
\item Dritter Teil der Aufzählung.
\end{enumerate}
```

Resultat

1. Erster Teil der Aufzählung. Er darf ruhig länger als eine Zeile sein.
2. Zweiter Teil der Aufzählung.
3. Dritter Teil der Aufzählung.

Aufzählungen II

- ▶ Die Symbole vor den einzelnen Teilen einer `enumerate`-Aufzählungen werden automatisch erzeugt.
- ! Man kann die automatisch erzeugten Symbole gegen frei wählbare auswechseln, indem man `\item` durch
$$\item[Symbol]$$
- ▶ Weitere Aufzählungen:
 - ▶ `itemize` hier sind die Symbole nicht frei wählbar
 - ▶ `description` für variable Listen

Beispiel:

```
\begin{enumerate}
\item Erster Textteil
  \begin{description}
    \item[zu 1:] Erster Unterteil
    \item[zu 2:] Zweiter Unterteil
  \end{description}
\item Zweiter Textteil
  \begin{enumerate}
    \item[a)] Erster Unterteil
    \item[b)] Zweiter Unterteil
  \end{enumerate}
\item Dritter Textteil
  \begin{itemize}
    \item Erster Unterteil
    \item Zweiter Unterteil
  \end{itemize}
\end{enumerate}
```

Resultat:

1. Erster Textteil

zu 1: Erster Unterteil

zu 2: Zweiter Unterteil

2. Zweiter Textteil

a) Erster Unterteil

b) Zweiter Unterteil

3. Dritter Textteil

- Erster Unterteil

- Zweiter Unterteil

Tabellen I

Eine einfache Tabelle mit zwei Spalten (links- bzw. rechtsbündig), aufgebaut mittels einer `tabular` -Umgebung:

```
\begin{tabular}{lr}  
  Sportart & Zeit \\  
  Laufen (100m) & 5 min \\  
  Schwimmen (50 m) & 30 min \\  
  Radfahren (1 km) & 20 min \\  
\end{tabular}
```

Resultat

Sportart	Zeit
Laufen (100m)	5 min
Schwimmen (50 m)	30 min
Radfahren (1 km)	20 min

Tabellen II

Die gleiche Tabelle mit zusätzlichen senkrechten und waagerechten Linien:

```
\begin{tabular}{|l|r|}  
 \hline  
 Sportart & Zeit \\ \hline \hline  
 Laufen (100m) & 5 min \\ \hline  
 Schwimmen (50 m) & 30 min \\ \hline  
 Radfahren (1 km) & 20 min \\ \hline  
 \end{tabular}
```

Resultat

Sportart	Zeit
Laufen (100m)	5 min
Schwimmen (50 m)	30 min
Radfahren (1 km)	20 min

Weitere Möglichkeiten bei Tabellen

- ▶ Zentrieren von Spalten durch das Symbol `c` statt `l` oder `r`
- ▶ Feste Spaltenbreite, z.B. von 3 cm bei der ersten Spalte, durch:

```
\begin{tabular}{p{3cm}r}
```

- ▶ Beliebiger Text vor, zwischen oder nach Spalten, z.B. ein Punkt zwischen erster und zweiter Spalte:

```
\begin{tabular}{l@{.}r}
```

- ▶ Zusammenfassen von mehreren Spalten eine Zeile zu einer Spalte, z.B. von 2 Spalten zu einer zentrierten Spalte mit dem Eintrag Zeit:

```
\multicolumn{2}{c}{Zeit}
```

- ▶ Waagerechte Linien nur von Spalte m bis n , z.B. von Spalte 2 bis 4:

```
\cline{2-4}
```

Das Setzen von mathematischen Texten I

Einige Beispielbefehle für mathematische Symbole:

Beschreibung	Befehl	Aussehen
Gr. Buchst.	<code>\pi</code>	$\pi \approx 3.14159$
Summe	<code>\sum</code>	$\sum_{i=1}^n i$
Wurzel	<code>\sqrt</code>	$\sqrt{a^2 + b^2}$
Integral	<code>\int</code>	$\int_a^b f(x) dx$
Quantor	<code>\forall</code> <code>\exists</code>	$\forall x \exists y : x^2 = y$
Bruch	<code>\frac{}{}</code>	$\frac{n(n+1)}{2}$
Grenzwert	<code>\lim</code>	$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$
Leere Menge	<code>\emptyset</code>	$M = \emptyset$
Vereinigung	<code>\bigcup</code>	$\mathcal{M} = \bigcup_{i \in \mathcal{I}} \mathcal{N}_i$
Pfeil	<code>\rightarrow</code>	$x = 0 \rightarrow x \geq 0$
Komposition	<code>\circ</code>	$(f \circ g)(x)$

Das Setzen von mathematischen Texten II

Mit dem Paket `amssymb` kann man auch kalligrafische, Fraktur- und „Blackboard“-Buchstaben schreiben:

$$\left. \begin{array}{l} \backslash\mathrm{cal}\{A\} \\ \backslash\mathrm{mathfrak}\{A\} \\ \backslash\mathrm{mathbb}\{A\} \end{array} \right\} \text{ bringt } \left\{ \begin{array}{l} \mathcal{A} \\ \mathfrak{A} \\ \mathbb{A} \end{array} \right.$$

In den Text eingebettete mathematische Formeln werden durch das Dollarsymbol geklammert

Beispiel

Es seien a und b die Katheten und c die Hypotenuse eines rechtwinkligen Dreiecks. Dann gilt $c = \sqrt{a^2 + b^2}$ (Satz des Pythagoras).

Resultat

Es seien a und b die Katheten und c die Hypotenuse eines rechtwinkligen Dreiecks. Dann gilt $c = \sqrt{a^2 + b^2}$ (Satz des Pythagoras).

Das Setzen von mathematischen Texten III

Vom Text durch Zentrierung abgesetzte mathematische Formeln werden durch die Klammerung `\[\]` erzielt:

Beispiel

Es seien a und b die Katheten und c die Hypotenuse eines rechtwinkligen Dreiecks. Dann gilt

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

(Satz des Pythagoras).

Resultat:

Es seien a und b die Katheten und c die Hypotenuse eines rechtwinkligen Dreiecks. Dann gilt

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

(Satz des Pythagoras).

Das Setzen von mathematischen Texten IV

Mit der Umgebung `equation` kann man zentrierte mathematische Formeln durchnummerieren und auch auf diese verweisen.

Beispiel

Es seien a und b die Katheten und c die Hypotenuse eines rechtwinkligen Dreiecks. Dann gilt

```
\begin{equation}
  c = \sqrt{a^2 + b^2}
  \label{eq:PHY}
\end{equation}
```

(Satz des Pythagoras). Eine Konsequenz von ([\ref{eq:PHY}](#)) ist ...

Resultat:

Es seien a und b die Katheten und c die Hypotenuse eines rechtwinkligen Dreiecks. Dann gilt

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

(Satz des Pythagoras). Eine Konsequenz von (1) ist ...

Das Setzen von mehrzeiligen Formeln

Für das Setzen von mehrzeiligen Formeln nutzt man die zur `tabular` -Umgebung analogen `array` -Umgebung.

Beispiel

```
\[
\begin{array}{rcl}
1 & = & 1+0 & \text{da 0 neutral} \\
& = & 1+0+0 & \text{da 0 neutral} \\
& = & 1+0+0+0 & \text{da 0 neutral} \\
& = & \dots &
\end{array}
\]
```

Resultat

1	=	1+0	da 0 neutral
	=	1+0+0	da 0 neutral
	=	1+0+0+0	da 0 neutral
	=	...	

Größenanpassung der Klammerung I

Die Größenanpassung an Formeln ist durch die Befehle `\left` und `\right` möglich.

Beispiel ohne Klammersanpassung

```
\[
M =
[
\begin{array}{cc}
1 + (\frac{1}{1-\frac{x}{2}})^3 & 0 \\
0 & 1 + (\frac{1}{1-\frac{x}{2}})^3
\end{array}
]
\]
```

Resultat

$$M = \begin{bmatrix} 1 + \left(\frac{1}{1-\frac{x}{2}}\right)^3 & 0 \\ 0 & 1 + \left(\frac{1}{1-\frac{x}{2}}\right)^3 \end{bmatrix}$$

Größenanpassung der Klammerung I

Das gleiche Beispiel mit einer Klammeranpassung:

```
\[
M =
\left[
\begin{array}{cc}
1 + \left(\frac{1}{1-\frac{x}{2}}\right)^3 & 0 \\
0 & 1 + \left(\frac{1}{1-\frac{x}{2}}\right)^3
\end{array}
\right]
\]
```

Resultat:

$$M = \begin{bmatrix} 1 + \left(\frac{1}{1-\frac{x}{2}}\right)^3 & 0 \\ 0 & 1 + \left(\frac{1}{1-\frac{x}{2}}\right)^3 \end{bmatrix}$$

Bilder einfügen

Mit dem Paket `graphicx` ist es möglich `jpg`, `png`, `bmp` und `pdf` Dateien einzufügen. Mit dem Befehl

```
\includegraphics[scale=0.5]{tuxnapping.jpg}
```

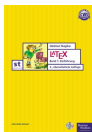
kann man ein Bild einfügen. `scale` gibt dabei den Skalierungsfaktor des Bildes an.



Und nun?

Es gibt noch vieles weitere in \LaTeX zu entdecken,
viel Spaß dabei... (-:

Literatur



Helmut Kopka: *LaTeX - Einführung Band 1*; Addison Wesley.



Christian Baun: *LaTeX für Dummies*; mitp-Verlag



Matthias K. Dalheimer: *LaTeX kurz & gut*; O'Reilly.

Links

- ▶ <http://www.miktex.de>
- ▶ <http://www.dante.de/>
Die wichtigste Seite, hier gibt es auch die deutschsprachige FAQ.
- ▶ <http://ad.informatik.uni-freiburg.de/bibliothek/tutorials/>
Hier gibt es ein paar nette Skripte zu \LaTeX
- ▶ <http://www.dante.de/TeX-Service-Paket/tex/cookbook/cookbook.html>
Kochbuch für \LaTeX
- ▶ <http://www.weinelt.de/latex/>
Befehlsübersicht