



Berufs Bildung Baden



L^AT_EX - Einführung

Die wichtigsten Schritte und BBB Spezialitäten

L^AT_EX

Angaben zum Dokument

Dateiname: Latex
Vorlage: $\text{\LaTeX} 2_{\epsilon}$
Verantwortliche(r): Hanspeter Vogt <hanspeter.vogt@bbb Baden.ch>
Michael Schneider <michael.schneider@bbb Baden.ch>
Builddatum: 29. Dezember 2006
Copyright: © BerufsBildungBaden, 2006

Änderungsnachweis

Datum	Version	Autor	Bemerkungen
Juli 2005	1.0	HPV	Erste Version
10.10.2005	2.0	MSC	Ergänzt mit \LaTeX Besonderheiten

Zusammenfassung

Dieser Text ist eine Zusammenfassung der wichtigsten Elemente beim Arbeiten mit \LaTeX . Er ist entstanden aus den Übungen, die ich als Einführung in \LaTeX durchgearbeitet habe. Es handelt sich keinesfalls um eine systematische Einführung in \LaTeX sondern vielmehr um eine Sammlung der wichtigen Dinge in der Reihenfolge, in der sie für mich relevant geworden sind.

Inhaltsverzeichnis

1	Links	1
1.1	L ^A T _E X-Umgebung für Windows	1
1.2	L ^A T _E X-Editor	1
1.3	L ^A T _E X-Erweiterungen	2
1.4	Tools	2
2	Arbeit mit der Software	3
2.1	Editoren	3
2.2	Installation von Packages	4
3	Grundlagen	5
3.1	Dokumentaufbau	5
3.2	Umbruch: Zeilen, Absatz, Seiten	6
3.3	Dokumentklassen und Packages	7
3.4	Kapiteleinteilung	8
3.5	Tabellen	8
3.6	Darstellung von Sonderzeichen	10
3.7	Formate	11
3.8	Schriftgrösse	12
3.9	Anführungszeichen	12
3.10	Fussnoten	13
3.11	Marginalien	13
3.12	Auflistungen	13
3.13	Erstellen von Verzeichnissen	16
3.14	Arbeiten mit Tabulatoren	16
3.15	Erstellen von Querverweisen	17
3.16	Quellenangaben	18
3.17	Titelseiten erstellen	19
3.18	URLs	19
4	Weiterführende Befehle	20
4.1	Teildokumente	20
4.2	Eigene Kommandos definieren	20
4.3	Variablen in L ^A T _E X 2 _ε	21
4.4	Fremdsprachen und Symbole	23
4.5	Mathematische Symbole	24

4.6	Mathematische Akzente	26
4.7	Rahmen und Striche	29
4.8	Nummerierung von Tabellen und Abbildungen	31
4.9	Kopf- und Fusszeilen	32
5	Hilfreiche Befehle	34
5.1	Integration von PS-Grafiken	34
5.2	Verdrehen von Objekten	34
5.3	Mehrspaltendruck	34
5.4	Noch viel mehr Schriften	35
5.5	Nützliche Symbole	37
5.6	Setzen von Listings	38
5.7	Zitate	40
6	PDF Spezialitäten	41
6.1	Dateien in PDF einbetten	41
7	Indices	43
7.1	Index	43
7.2	Glossar	44
8	Grafiken	45
8.1	Grafikformate	45
8.2	Grafiken als Figures/Floats	45
8.3	Grafiken im Text	46
8.4	Grafiken mit PGF	48
A	BBB-Standards	54
A.1	SVN Verzeichnisstruktur	54
A.2	L ^A T _E X Verzeichnisstruktur	54
A.3	Schreibweisen	55
A.4	Allgemeines	55
B	BBB-Styles	56
B.1	Prüfungen erstellen mit <code>bbb-new-test.sty</code>	56
	Literaturverzeichnis	62
	Glossar	63
	Index	64

Kapitel 1

Links

Hier sind die Links zusammengestellt, die für die Installation der \LaTeX Umgebung und den Download eines Editors nötig sind.

1.1 \LaTeX Umgebung für Windows

Als Umgebung für \LaTeX arbeiten wir mit MikTeX. Die Umgebung kann unter folgendem Link heruntergeladen werden:

<http://www.miktex.org/setup.html> Danach einfach Setup starten und mit den Standardwerten eine Vollinstallation durchführen.

1.2 \LaTeX Editor

1.2.1 Windows

TexNic (Freeware)

Ein einfach und intuitiv zu bedienender Editor, der zudem noch als Freeware vertrieben wird, ist TexNic. Er kann unter folgendem Link heruntergeladen werden:

http://www.texniccenter.org/front_content.php?idcat=50

WinEdt (Shareware)

Ein mächtiger Editor - unter anderem mit Rechtschreibkontrolle - ist WinEdt. Er kann unter folgendem Link heruntergeladen werden:

<http://www.winedt.com/>

Nach der Testphase kostet er ca. Fr. 30.–.

Weitere Hinweise zur Arbeit mit den Editoren finden sich im gleichnamigen Kapitel.

1. Links

1.2.2 Linux

Kile

Unter Linux hat sich Kile sehr bewährt. Kile ist eine KDE-Applikation mit vielen Goodies.

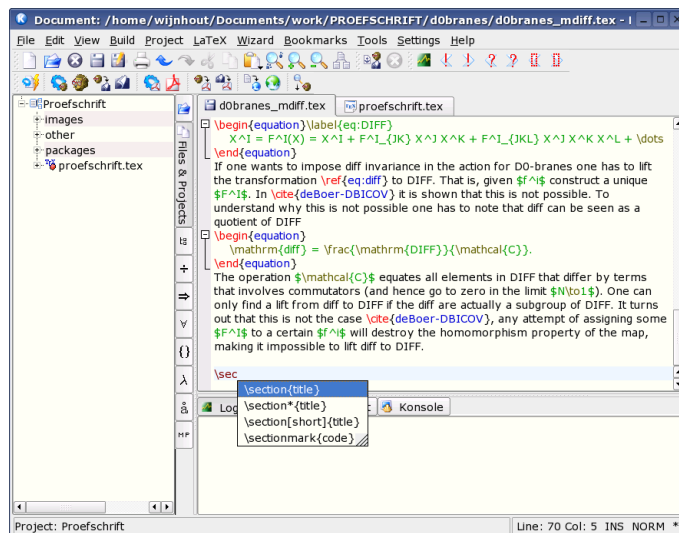


Abbildung 1.1: Kile ist ein \LaTeX Editor für Linux

1.3 \LaTeX Erweiterungen

1.3.1 CTAN – Comprehensive \TeX Archive Network

Obwohl unscheinbar ist CTAN *die* zentrale Anlaufstelle für \LaTeX Packages und Klassen. Zu finden unter <http://www.ctan.org>.

1.4 Tools

Chktex ist ein böser, pingelige \LaTeX Quellcodechecker. Sehr interessant und niederschmetternd <http://baruch.ev-en.org/proj/chktex/>.

Kapitel 2

Arbeit mit der Software

2.1 Editoren

2.1.1 WinEdt

Bildschirmumbruch

Allein schon das Einstellen des Bildschirmumbruchs in WinEdt hat mich zwei Stunden gekostet. Folgendermassen kann es erreicht werden:

„In this case, enable the conventional Soft Wrapping option in the Wizard. And to ensure your TeX documents are also subject to this method of wrapping, insert the string TeX; at the front of the mode filter associated with the option Conventional (Soft) Wrapping in this dialog. That is, change the default filter in this dialog from:

```
HTML;ANSI;Soft|Hard
```

to

```
TeX;HTML;ANSI;Soft|Hard“
```

Projekttree erzeugen

Damit der Projekttree erzeugt werden kann, muss das Hauptdokument markiert werden. Darauf kann im Menü Project der Befehl Set Main File angeklickt werden. Danach kann im Projektbaumfenster durch Anklicken der entsprechenden Schaltfläche der Baum erzeugt werden.

2.1.2 TeXnicCenter

WinEdt ist ein Shareware-Editor, der 30 Tage lang getestet werden kann. Die Arbeit mit WinEdt ist gewöhnungsbedürftig, da er sehr viele Möglichkeiten bietet.

Als einfachere und günstigere Alternative bietet sich ein Freeware-Editor an: *TeXnicCenter*

von der Firma ToolsCenter: www.toolscenter.org. Dieser für Einsteiger in LaTeX ausserordentlich gut geeignete Editor bietet ein intuitives Benutzerinterface, das mit Symbolleisten ein einfaches Arbeiten im Bereich der Formatierung und insbesondere der Erstellung mathematischer Formeln bietet.

Der *Projekttree* wird in TeXnicCenter folgendermassen erzeugt: Projekt \Rightarrow Erzeugen mit aktueller Datei als Hauptdatei

TeXnicCenter bietet eine *sehr gute Hilfe* zu LaTeX allgemein, wohingegen die Hilfe in WinEdt in vielen Bereichen eher kryptisch gehalten ist.

Ein Nachteil liegt darin, dass nicht eine Datei, sondern jeweils nur die Hauptprojektdatei in ein Ausgabeformat umgewandelt werden kann, was zu einem Klick mehr führt.

2.2 Installation von Packages

2.2.1 Allgemein

Normalerweise erhält man für ein Package eine DATEINAME.INS und/oder eine DATEINAME.DTX Datei. Ist in der Datei README nichts anderes beschrieben, müssen diese Dateien mit \LaTeX bearbeitet werden: `latex dateiname.ins` und/oder `latex dateiname.dtx` extrahiert die Dokumentation und verschiedene Stylefiles mit der Endung .STY.

Details über die Installation finden sich unter <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=instpackages>.

Um dieses Package benutzen zu können, trennen sich hier die Wege je nach Betriebssystem.

2.2.2 Unter Debian-Linux

Um ein Package unter Debian Linux zu installieren muss es irgendwo in `/USR/SHARE/TEXMF/F/TEX/LATEX/` untergebracht werden. Es können auch eigene Verzeichnisse verwendet werden. Das Kommando `texhash` erstellt die Datenbank, und ab da ist es benutzbar.

Kapitel 3

Grundlagen

L^AT_EX ist ein ausserordentlich flexibles, rechner- und betriebssystemunabhängiges Satzsystem, das zur Erstellung von Dokumenten in Buchdruckqualität geeignet ist. Es trennt den Inhalt und die Formatierung.

3.1 Dokumentaufbau

Dokumente in L^AT_EX sind immer folgendermassen aufgebaut.

```
\documentclass{article}
\begin{document}
\begin{center}
\Large
Der Herr der Ringe
\tiny
von J.R.R. Tolkien
\end{center}
```

Die Hobbits sind ein unauffälliges, aber sehr altes Volk, das früher zahlreicher war als heute; denn sie lieben Frieden und Stille und einen gut bestellten Boden: eine wohlgeordnete und wohlbewirtschaftete ländliche Gegend war ihr bevorzugter Aufenthaltsort.

```
\end{document}
```

Das ergibt folgende Ausgabe:

Der Herr der Ringe von J.R.R. Tolkien

Die Hobbits sind ein unauffälliges, aber sehr altes Volk, das früher zahlreicher war als heute; denn sie lieben Frieden und Stille und einen gut bestellten Boden: eine wohlgeordnete und wohlbewirtschaftete ländliche Gegend war ihr bevorzugter Aufenthaltsort.

3.2 Umbruch: Zeilen, Absatz, Seiten

\LaTeX optimiert die Seite für das Lesen, d.h. es verwendet Boxen, in denen die Abstände augenfreundlich dargestellt werden. Um \LaTeX zu unterstützen können Trennhilfen (bedingte Trennstriche) bei Fremdwörtern eingegeben werden, die nur am Zeilenende wirklich erscheinen.

.....Arche\opterix.....

Wörter können über Leeräume auch verklebt werden.

siehe Seite\~7 *ergibt* siehe Seite 7

Im Textsatz unterscheiden sich die Abstände zwischen den Worten und nach einem Punkt. Bei Abkürzungen muss man \LaTeX etwas helfen und vor den Leerschlag ein Backslash machen. Dies fällt besonders bei mehrzeiligen Texten auf:

Dieser Leerschlag ist mit und ohne Backslash, d.h. hier ist der Abstand schön oder nichtbeim mehrzeiligen Texten.

Dieser Leerschlag ist mit und ohne Backslash, d.h. hier ist der Abstand schön oder nichtbeim mehrzeiligen Texten.

(Dieser Leerschlag ist mit und ohne Backslash, d.h.\ hier ist der Abstand schön oder nicht)

Erzwungene Zeilenumbrüche werden mit `\` erzeugt; dahinter kann in `[0.5 cm]` der Abstand angegeben werden.

Er könnt aus lauter Lust am Prassen\
 Beefsteaks aus Gold sich braten lassen\ [0.5 cm]
 Doch darf er eines nicht vergessen\
 Beefsteaks aus Gold kann man nicht essen.\ [0.5 cm]

Er könnt aus lauter Lust am Prassen
 Beefsteaks aus Gold sich braten lassen

Doch darf er eines nicht vergessen
 Beefsteaks aus Gold kann man nicht essen.

3.2.1 Seitenumbrüche

Einen Seitenumbruch erzeugt man mit `\newpage`. Noch besser allerdings ist `\clearpage`, welches zuerst alles erledigt, das noch gemacht werden muss: Floats platzieren, etc...

Soll der Seitenumbruch in einem Textabschnitt verhindert werden, verwendet man `\begin{samepage} ... \end{samepage}`.

3.2.2 Absätze ausrichten

LaTeX setzt den Text standardmässig in Blocksatz mit den oben erwähnten Ausgleichen. Text kann aber auch anders ausgerichtet werden.

Ich will	Schlüsselwort
zentrieren	<code>\begin{center} ... \end{center}</code>
einzelne Zeile	<code>\centerline{text}</code>
rechter Flatterrand	<code>\begin{flushleft} ... \end{flushleft}</code>
linker Flatterrand	<code>\begin{flushright} ... \end{flushright}</code>
einzelne Zeile	<code>\raggedleft{text}</code> oder <code>\raggedright{text}</code>

Einrückungen beginnen mit `\begin{quote}` und enden mit `\end{quote}`

3.3 Dokumentklassen und Packages

3.3.1 Dokumentklassen

Die Dokumentklasse legt das grundsätzliche Layout eines Dokumentes fest; es ist das erste Kommando einer Datei, das in geschweiften Klammern die Dokumentenklasse festlegt.

Untenstehend eine Auswahl der möglichen Klassen:

Klasse	Verwendung
book	für komplette Bücher
report	für umfangreiche Dokumente
article	für mittelgrosse und kleine Dokumente

Gleichzeitig lässt sich mit Hilfe von Parametern in eckigen Klammern das Textlayout beeinflussen.

Auswahl von Parametern:

Parameter	Wirkung
11pt	Vergrößerung der Grundschrift um 10 Prozent
12pt	Vergrößerung der Grundschrift um 20 Prozent
twoside	Erstellen zweiseitiger Dokumente
twocolumn	zweispaltige Ausgabe
a4paper	Grössenanpassung an DIN A4-Format

Zum Beispiel:

```
\documentclass[11pt,twoside,a4paper]{book}
```

3.3.2 Packages

Packages werden benötigt, damit ganz spezielle Aufgaben erledigt werden können. Sie werden mit folgendem Befehl eingebunden:

```
\usepackage[optionale Parameter]{packagename}
```

3.4 Kapiteleinteilung

Die Einteilung in Kapitel und Unterkapitel funktioniert auch mit \LaTeX Befehlen:

```
\section{...} --> 1
\subsection{...} --> 1.1
\subsubsection{...} --> 1.1.1
\patagraph{...}
\subparagraph{...}
```

Soll ein Text in Bereiche Teil 1, Teil 2 usw. eingeteilt werden, hilft der Befehl `\part{...}`. Für die Dokumentklasse *book* und *report* gibt es noch eine übergeordnete Einteilung in `\chapter{...}`.

3.5 Tabellen

3.5.1 Normale Tabellen

Aufbau

Tabellen können mit \LaTeX folgendermassen erzeugt werden. Das Arbeiten mit Tabulatoren finde ich mühsamer, es wird in Kapitel 3.14 auf Seite 16 beschrieben:

```
\begin\{tabular\}\{l11\}\
Chicago \& U.S.A. \& 1893\
Zürich \& Switzerland \& 1897\
Paris \& France \& 1900\
Heidelberg \& Germany \& 1904\
Rome \& Italy \& 1908\
\end\{tabular\}\[0.5 cm]
```

Das ergibt folgende Ausgabe:

Chicago	U.S.A.	1893
Zürich	Switzerland	1897
Paris	France	1900
Heidelberg	Germany	1904
Rome	Italy	1908

3. Grundlagen

Folgende Spaltenmerkmale können definiert werden:

l	linksbündig ausgerichtete Spalte
r	rechtsbündig ausgerichtete Spalte
c	zentrierte Spalte
	ein senkrechter Strich in Tabellenhöhe

Innerhalb der Tabelle gilt folgendes:

- & trennt die einzelnen Spalten
- \\ trennt die einzelnen Zeilen
- \hline zieht einen waagrechten Strich über die ganze Seite.

Texte in Tabellen

Will man Texte, die Umbrechen in Tabellen schreiben, so benötigt man die Anweisung `p{}`.

Beispiel:

Dieser Text bricht nicht um!	Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text! Dies ist ein langer, umbrechbarer Text!
------------------------------	---

```

\begin{tabular}{|l|p{4cm}|}
\hline
Dieser Text bricht nicht um! &
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
  Dies ist ein langer, umbrechbarer Text!
\hline
\end{tabular}

```

3.5.2 Tabellen mit fester Breite

Will man eine Tabelle mit fester Breite erstellen, so empfiehlt sich `tabularx`. Dabei ist das erste Argument die Breite und ein Bezeichner `X` bezeichnet dehnbare Spalten

Eine normale Spalte	Diese Spalte dehnt sich.	Eine normale Spalte
---------------------	--------------------------	---------------------

```

\begin{tabularx}{\textwidth}{|l|X|l|}
\hline Eine normale Spalte & Diese Spalte dehnt sich. & Eine normale Spalte \\
\end{tabularx}
  
```

3.5.3 Tabellen über mehrere Seiten

Longtable? Todo !

3.5.4 Einfärben von Zellen

Das Einfärben von ganzen Zeilen geschieht mit `\rowcolor`. Einzelne Zellen können mit `multicolumn` eingefärbt werden. Beispiel:

Nmormale	Zeile	
Eingefärbte	Zeile	
Eingefärbte	Zelle	

```

\begin{tabularx}{\textwidth}{|l|l|l|}
\hline Normale & Zeile & \\
\hline \rowcolor[gray]{.8}[.95\tabcolsep] Eingefärbte & Zeile & \\
\hline
\end{tabularx}
  
```

3.6 Darstellung von Sonderzeichen

Die folgenden Zeichen können in \LaTeX nicht direkt dargestellt werden; die nachstehenden Zeichenkombinationen geben sie trotzdem aus:

gewünschtes Zeichen	einzugebendes Zeichen
{	\{
}	\}
#	\#
&	\&
-	_
%	\%
\$	\\$
\	\$\$\backslash\$
^	\^{}
~	\~{}

Alternativ kann für einen bestimmten Textabschnitt, der genau so ausgegeben werden soll, wie er eingegeben worden ist, die verbatim-Umgebung eingeschaltet werden, die verhindert, dass Umbrüche durch- und \LaTeX Kommandos ausgeführt werden. Ist nur ein kleines Textstück betroffen, hilft der `\verb`-Befehl:

```
\verb=          TEST          = fertig
          TEST          fertig
```

Die Verbatim Umgebung:

```
Blah Bla \section{Befehle}
  Uhganna uhganna
  foobar di trallalla
$ _ \ $ %
```

```
\begin{verbatim}
Blah Bla \section{Befehle}
  Uhganna uhganna
  foobar di trallalla
$ _ \ $ %
\end{verbatim}
```

3.7 Formate

Es lassen sich verschiedene Formatierungen vornehmen. An der BBB IT-School werden aber nur sehr wenige eingesetzt, damit die Texte übersichtlich bleiben.

Schlüsselwort	Ausgabe
<code>\emph</code>	<i>Hervorhebung</i>
<code>\texttt</code>	für Eingabe in ein System \Rightarrow nicht proportionale Schrift

Der Vorteil von `emph{}` ist das intelligente Hervorheben. In einem normalen Text wird *kursiv* hervorgehoben, *in einem kursiven Text passiert die Hervorhebung dementsprechend anders.*

Weitere Möglichkeiten:

3. Grundlagen

Schlüsselwort	Ausgabe
<code>\textbf{boldface oder fett}</code>	Hervorhebung als boldface oder fett
<code>\textsl{kursiv}</code>	<i>kursiv</i>
<code>\textup{upright}</code>	upright
<code>\textit{italic}</code>	<i>italic</i>
<code>\textsc{SmallCaps}</code>	SMALLCAPS
<code>\textbf{\textsl{fett und kursiv}}</code>	<i>fett und kursiv</i>
<code>\textmd{medium}</code>	medium
<code>\textrm{roman}</code>	roman
<code>\textsf{sansserif}</code>	sansserif
<code>\texttt{typewriter}</code>	typewriter

3.8 Schriftgrösse

Schlüsselwort	Ausgabe
<code>\tiny{winzig}</code>	winzig
<code>\scriptsize{sehr klein}</code>	sehr klein
<code>\footnotesize{ziemlich klein}</code>	ziemlich klein
<code>\small{klein}</code>	klein
<code>\normalsize{normalgross}</code>	normalgross
<code>\large{etwas grösser}</code>	etwas grösser
<code>\Large{ziemlich gross}</code>	ziemlich gross
<code>\LARGE{gross}</code>	gross
<code>\huge{sehr gross}</code>	sehr gross
<code>\Huge{riesig}</code>	riesig

3.9 Anführungszeichen

Anführungszeichen sind nicht ganz einfach auszugeben; eine Übersicht bietet die folgende Tabelle:

Tastatureingabe	Druckausgabe	Erklärung
<code>‘ ‘</code>	“	amerikanische Anführungszeichen links oben
<code>’ ’</code>	”	amerikanische Anführungszeichen rechts oben
<code>“ “</code>	„	deutsche Anführungszeichen links unten
<code>” ”</code>	“	deutsche Anführungszeichen rechts oben
<code>-</code>	-	Bindestrich
<code>--</code>	—	n-Strich
<code>---</code>	—	m-Strich
<code>\$\$</code>	—	Minuszeichen im Mathematikmodus

3.10 Fussnoten

Fussnoten werden am Ende jeder Seite verwaltet und standardmässig kapitelweise nummeriert. Der Text wird nach dem Befehl `\footnote{in geschweifte Klammern geschrieben}`¹. Fussnoten können sich über Fussnoten mehrere Seiten erstrecken² oder mathematische Formeln enthalten.

3.11 Marginalien

Randbemerkungen oder Marginalien werden mit `=\marginpar{Randbemerkung}=`³ erzeugt.

3.12 Auflistungen

3.12.1 Spiegelstrichlisten

Spiegelstrichlisten werden durch die `itemize`-Umgebung ermöglicht.

```
\begin{itemize}
\item erster Punkt
\item zweiter Punkt
\item dritter Punkt
\item ...
\item letzter Punkt\\
\end{itemize}
```

- erster Punkt
- zweiter Punkt
- dritter Punkt
- ...
- letzter Punkt

Die Aufzählung kann bis in 4 Ebenen vorgenommen werden; jede Ebene wird von einem `=\begin{itemize}=` und `=\end{itemize}=` umschlossen.

3.12.2 Nummerierte Liste

Sie ähnelt der Spiegelstrichliste sehr, beginnt aber mit einem anderen Befehl:

¹Verwendung des Fussnotenbefehls

²wie in Rechtstexten üblich

³siehe Beispiel [3.10 13](#)

```

\begin{enumerate}
\item erster Punkt
\item zweiter Punkt
\item dritter Punkt
\item ...
\item letzter Punkt\
\end{enumerate}
  
```

1. erster Punkt
2. zweiter Punkt
3. dritter Punkt
4. ...
5. letzter Punkt

3.12.3 Definitionsliste

Definitionslisten zeigen den zu definierenden Begriff fett, dahinter folgt eingerückt die Definition für den Begriff.

```

\begin{description}
\item[Spiegelstrichlisten] Listen, bei welchen den einzelnen Aussagen
Gliederungspunkte mit negativem Erstzeileneinzug vorangestellt sind.
\item[Nummerierte Listen] Listen, bei welchen den einzelnen Aussagen
Nummern mit negativem Erstzeileneinzug vorangestellt sind.
\item[Definitionslisten] Listen, bei welchen den einzelnen Aussagen
die zu definierenden Begriffe mit negativem Erstzeileneinzug und fett
gedruckt vorangestellt sind.
\end{description}
  
```

Das ergibt folgende Ausgabe:

Spiegelstrichlisten Listen, bei welchen den einzelnen Aussagen Gliederungspunkte mit negativem Erstzeileneinzug vorangestellt sind.

Nummerierte Listen Listen, bei welchen den einzelnen Aussagen Nummern mit negativem Erstzeileneinzug vorangestellt sind.

Definitionslisten Listen, bei welchen den einzelnen Aussagen die zu definierenden Begriffe mit negativem Erstzeileneinzug und fett gedruckt vorangestellt sind.

Bei allen Listen können die Aufzählungszeichen im Format angepasst werden. Dies sollte allerdings immer im Rahmen einer eigenen Umgebung passieren, da sonst die Listen im ganzen Dokument so geändert sind.

Beispiel:

```

1 \newenvironment{myDescription}
2   {\begin{list}}{\let\makelabel\myDescriptionlabel
  
```

3. Grundlagen

```

3     \setlength\labelwidth{40pt}%
4     \setlength\leftmargin{\labelwidth+\labelsep}}}%
5     {\end{list}}
6 \newcommand{\myDescriptionlabel}[1]{\textbf{\emph{\textsf{#1}}}}
7
8 \begin{myDescription}
9 \item[Spiegelstrichlisten] Listen, bei welchen den einzelnen Aussagen
10 Gliederungspunkte mit negativem Erstzeileneinzug vorangestellt sind.
11 \item[Nummerierte Listen] Listen, bei welchen den einzelnen Aussagen
12 Nummern mit negativem Erstzeileneinzug vorangestellt sind.
13 \item[Definitionslisten] Listen, bei welchen den einzelnen Aussagen
14 die zu definierenden Begriffe mit negativem Erstzeileneinzug und fett
15 gedruckt vorangestellt sind.
16 \end{myDescription}

```

Spiegelstrichlisten Listen, bei welchen den einzelnen Aussagen Gliederungspunkte mit negativem Erstzeileneinzug vorangestellt sind.

Nummerierte Listen Listen, bei welchen den einzelnen Aussagen Nummern mit negativem Erstzeileneinzug vorangestellt sind.

Definitionslisten Listen, bei welchen den einzelnen Aussagen die zu definierenden Begriffe mit negativem Erstzeileneinzug und fett gedruckt vorangestellt sind.

3.12.4 Spiegelstrichlisten in Tabellen

Da Spiegelstrichlisten in Tabellen etwas hässlich aussehen, besitzen wir an der BBB die `tabitemize` Umgebung. Sie übernimmt als Argument die Breite, die sie benutzen darf:

Hässlich	<ul style="list-style-type: none"> • Eine • etwas • sehr • hässliche • Liste
Schön	<ul style="list-style-type: none"> • Eine • etwas • sehr • schöne • Liste
Quelle der Hässlichkeit:	<pre> \begin{itemize} \item Eine \item ... \end{itemize} </pre>
Quelle der Schönheit:	<pre> \begin{tabitemize}{0.45\linewidth} \item Eine \item ... \end{tabitemize} </pre>

3.13 Erstellen von Verzeichnissen

3.13.1 Inhaltsverzeichnis

Das Inhaltsverzeichnis wird ausgehend von den Kapitelüberschriften des Dokumentes erstellt. Will man im Inhaltsverzeichnis einen anderen Text als in der Überschrift, kann das mit folgendem Befehl erreicht werden:

```
\section[Titel im Inhaltsverzeichnis]{Titel}
```

Das Inhaltsverzeichnis wird mit folgendem Befehl erstellt:

```
\tableofcontents
```

Soll ein Titel nicht im Inhaltsverzeichnis erscheinen, hilft folgender Befehl:

```
\section*{Titel}
```

3.13.2 Tabellenverzeichnis

Ein Tabellenverzeichnis wird mit `\listoftables` erstellt. (Verwaltet in `*.lot.`)

3.13.3 Abbildungsverzeichnis

Ein Abbildungsverzeichnis wird mit `\listoffigures` erstellt. (Verwaltet in `*.lof.`)

3.13.4 Anhänge

Anhänge werden mit dem Befehl `\appendix` über die normalen `\section`-Befehle erzeugt und verwaltet.

3.14 Arbeiten mit Tabulatoren

Das Arbeiten mit Tabellen ist bereits in Kapitel [3.5.1](#) auf Seite [8](#) erklärt worden; hier sei der Vollständigkeit halber noch das Arbeiten mit Tabulatoren gezeigt.

Innerhalb der Umgebung

```
\begin{tabbing}  
...  
und  
...  
\end{tabbing}
```

3. Grundlagen

wird mit `\=` eine Tabulatorstelle gesetzt und mit `\>` eine Tabulatorstelle angesprungen. Die einzelnen Zeilen werden wie bei den Tabellen durch `\\` getrennt. Neue Tabulatoren können jederzeit gesetzt werden; sie überschreiben die alten.

Folgende Artikel sind im Sortiment:

```
\begin{tabbing}
\hspace{5 cm} \= \kill
\> Artikel\\
\> Birne\\
\> Apfel\\
\end{tabbing}
```

Folgende Artikel sind im Sortiment:

```
Artikel
Birne
Apfel
```

3.15 Erstellen von Querverweisen

3.15.1 Verweise im Dokument

Querverweise ermöglichen es, bestimmte Stellen im Dokument zu referenzieren. Dazu muss an der zu referenzierenden Stelle ein Label (eine Bezeichnung) gesetzt werden. Diese Labels können an anderen Orten mit Kapitel- oder Seitenangabe aufgerufen werden.

Folgende Befehle sind in diesem Zusammenhang wichtig (Beispiel: Das Kapitel [3.14](#) auf der vorherigen Seite).

Befehl	Auswirkung	Beispiel
<code>\label{name}</code>	Setzt die Bezeichnung für einen Abschnitt. Dabei haben sich folgende Prefixe eingebürgert: sec: für Kapitel fig: für Floats (Figures) tab: für Tabellen eq: für Gleichungen aufgabe: für Aufgaben	<code>\label{sec:foobar}</code>
<code>\ref{name}</code>	Bezieht sich auf das Label mit der Kapitelangabe.	Siehe Kapitel 3.14 .
<code>\pageref{name}</code>	Bezieht sich auf das Label mit der Seitenangabe.	Siehe Seite 16 .
<code>\vref{name}</code>	Varioref: Schöne Referenzen.	Siehe Kapitel 3.14 auf der vorherigen Seite.

3.15.2 Verweise auf externe Dokumente

Das Package XR-HYPER erlaubt Verweise auf externe Dokumente (Siehe auch <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=extref>). Dafür braucht man die .AUX Datei des Dokumentes, wo man hinverweisen will. Zu Beginn braucht es die Anweisung:

```
\externaldocument[V1-]{volume1}[http://mybook.com/volume1.pdf]
```

Wobei

V1-: Ein Prefix ist, damit gleichnamige Labels keine Probleme ergeben.

volume1: Für den Dateinamen steht (VOLUME1.AUX).

http://mybook.com/volume1.pdf: Optional für eine URL steht, wo man dieses PDF erhält.

Hier ein Beispiel:

```
\externaldocument[226-]{extref/226_Objektorientiert_Implementieren_Theorie}
 [https://bbb-linux1.bbb.../226_Objektorientiert_Implementieren_Theorie.pdf]
```

UML Kollaborationsdiagramme sind unter Anderem im Dokument

```
\bbbfile{226\_Objektorientiert\_Implementieren\_Theorie.pdf}
 in Kapitel \vref{226-sec:kollaborationsdiagramme} beschrieben.
```

UML Kollaborationsdiagramme sind unter Anderem im Dokument 226-OBJEKTORIEN-TIERT-IMPLEMENTIEREN-THEORIE.PDF in Kapitel 3.4 auf Seite 31 beschrieben.

3.16 Quellenangaben

Quellenangabe ist Ehrensache. Mit L^AT_EX können ganze Literaturdatenbanken geführt werden. Es gibt allerdings auch eine einfachere Möglichkeit. Im Grunddokument muss irgendwo eine thebibliography umgebung definiert sein. Diese besitzt eigene Einträge wie folgt:

```
\begin{thebibliography}{99}
\bibitem{bib:mittenbach-goossens-2004} Mittenbach, Franz und Goossens, Michael:
  {\em The \LaTeX\ Companion - Second Edition},
  Addison Wesley, (ISBN 0-201-36299-6), Boston, 2004

\bibitem{bib:laermorama, 2004} Thomas Gastberger u.\ A.,
  {\em Lärmorama}, \url{http://www.laermorama.ch/},
  Fachstelle Lärmschutz, Kanton Zürich, 2004
```

[...]

```
\end{thebibliography}
```

Nun kann mittels des Kommandos cite darauf zugegriffen werden. Beispielsweise:

```
\cite{bib:mittenbach-goosens-2004} sagt ganz deutlich,  
dass keine Kommas oder Leerstellen verwendet werden sollten.
```

wird zu

[4] sagt ganz deutlich, dass keine Kommas oder Leerstellen verwendet werden sollten.

Die Bibliografie wird am Schluss dem Dokument angefügt.

3.17 Titelseiten erstellen

Zum Erstellen von Titelseiten stellt LaTeX einige spezielle Kommandos zur Verfügung.

```
\title{Einführung in \LaTeX\ \\  
Einführung \\  
Die ersten Kommandos}  
\author{H. Vogt}  
\thanks{M. Schneider}  
\date{Heute ist der \today}  
\maketitle
```

Das ergibt eine diesem Dokument ähnliche Titelseite.

3.18 URLs

Ein normaler Weblink kann mit dem Befehl `\url{http://...}` gesetzt werden: <http://www.ctan.org>. Für komplexere URLs, wie klickbare Mailadressen, braucht man den etwas komplexeren `\htmladdnormallink{ANZEIGE}{LINK}`.

Beispiel:

```
\htmladdnormallink{michael.schneider@bbb Baden.ch}  
{mailto:michael.schneider@bbb Baden.ch}
```

ergibt michael.schneider@bbb Baden.ch.

Kapitel 4

Weiterführende Befehle

4.1 Teildokumente

Umfangreiche Dokumente können in einzelne Kapitel aufgeteilt werden. In der Regel beginnen sie dann mit folgenden Befehlen:

```
\chapter{Weiterführende Befehle}  
\section{Teildokumente}
```

Die Ausgabe ist oben ersichtlich!

Im Hauptdokument, in dem die allgemeinen Angaben zum Dokument enthalten sind, wird mit folgenden Befehlen auf die einzelnen Kapitel verwiesen:

```
\input{sec/WinEdt.tex}  
\input{sec/Grundlagen.tex}  
\input{sec/Aufbau.tex}
```

In den eingebundenen Dokumenten dürfen die Befehle `\documentclass`, `\begin{document}` und `\end{document}` **nicht** enthalten sein.

4.2 Eigene Kommandos definieren

Mit eigenen Kommandos kann Tipparbeit gespart werden, z.B:

```
\newcommand{befehlname}{abzuarbeitende \LaTeX\-Befehle}
```

```
\newcommand{\w}{Wort }
```

```
\newcommand{wort}{\textsl{\huge Wort }}}
```

Dies ist ein kleines `\w` und dies ist eine riesiges `\wort`,
dies ist wieder ein kleines `\w` und wieder ein riesiges `\wort`.

Das ergibt als Ausgabe

Dies ist ein kleines Wort und dies ist eine riesiges *Wort* , dies ist wieder ein kleines Wort und wieder ein riesiges *Wort* .

4.3 Variablen in L^AT_EX 2_ε

Es werden die einfachen Variablen für Boolean, String und Integer vorgestellt.

4.3.1 Deklaration und Wertzuweisung

Typ	Deklaration	Wertzuweisung	Inhalt
Boolean	<code>\newboolean{Name}</code>	<code>\setboolean{Name}{Wahrheitswert}</code>	"true" oder "false"
String	<code>\newcommand{Name}</code>	<code>\def{Name}{String}</code>	String
Integer	<code>\newcounter{Name}</code>	<code>\setcounter{Name}{Integer}</code>	Ganzzahl

Die mit `\newcounter` deklarierte Integervariable ist ein normaler L^AT_EX -Zähler. Alle "counter"-Operationen wie `\value`, `\addtocounter`, `\stepcounter`, usw. können auf ihn angewandt werden.

Ein Beispiel

```
% Variablendeklaration
  \newboolean{farbeBlau}
  \newcommand\vorName\ % Leerzeichen "\ " notwendig!
  \newcounter{zahl}
% Wertzuweisung
  \setboolean{farbeBlau}{true}
  \def\vorName{Ernst}
  \setcounter{zahl}{2}
% Drucken der Variableninhalte
  \verb"vorName" hat den Inhalt ``\vorName''\
  \verb"zahl" hat den Inhalt ``\thezahl''\
```

Das ergibt folgende Ausgabe:

```
vorName hat den Inhalt "Ernst"
zahl hat den Inhalt "2"
```

4.3.2 Arithmetik mit dem Packet "calc"

Einfache Arithmetik kann mit T_EX-low-level Befehlen wie `\advance`, `\multiply`, usw. ohne weitere Massnahmen durchgeführt werden. Für erweiterte Berechnungen gibt es das Packet "calc" (`\usepackage{calc}`).

Für weitergehende Ausführungen sei wie immer auf die entsprechende Dokumentation verwiesen („The calc Package - Infix notation arithmetic in L^AT_EX“).

Ein Beispiel:

```

\newcounter{x}\newcounter{y}
\setcounter{x}{2}$x = \thex$\
\setcounter{y}{3}$y = \they$\
\setcounter{y}{(\value{x}+5)*\value{y}}
$y\rightarrow(x+5)\cdot y = \they$\

```

Das ergibt folgende Ausgabe:

$x = 2$

$y = 3$

$y \rightarrow (x + 5) \cdot y = 21$

4.3.3 Verwendung von Variablen mit der Kontrollstruktur "ifthen"

Das Packet "ifthen" muss eingebunden werden (`\usepackage{ifthen}`).

Befehl: `\ifthenelse{test}{then-Teil}{else-Teil}`

Ist die Bedingung `test` wahr, so wird der `then-Teil` ausgeführt und sonst der `else-Teil`.

Um eine String-Variable zu testen verwendet man das Makro `\equal`:

```
\equal{string1}{string2}
```

Wenn `string1` und `string2` identisch sind, ergibt `\equal` "true" und sonst "false".

Beispiele:

```

% Test von "meinName=Ernst"
\newcommand\meinName\
\def\meinName{Ernst}
\ifthenelse{\equal{\meinName}{Ernst}}
  {O.K. \meinName ist mein Vorname}
  {Nein! Ich heisse nicht \meinName}\

```

Das ergibt folgende Ausgabe:

O.K. Ernst ist mein Vorname

```

% Test einer Integervariable
\newcounter{z}\setcounter{z}{4}
$z=\thez$ und somit
\ifthenelse{\value{z}<3}
  {kleiner als 3.}
  {grösser oder gleich 3.}

```

4. Weiterführende Befehle

Das ergibt folgende Ausgabe:

$z = 4$ und somit grösser oder gleich 3.

4.3.4 Schleifenbildung

Mit dem Packet "ifthen" und einer Integervariable lassen sich sehr einfach `while-do`-Schleifen bilden.

Befehl: `\whiledo{test}{do}`

Es handelt sich um eine kopfgesteuerte Schleife. Solange die Auswertung von `test` "true" ergibt, wird der Schleifenrumpf `do` ausgeführt. Der Schleifenzähler kann einfach mit der "Counter"-Operation `\stepcounter{name}` um 1 inkrementiert werden. Andere Konstrukte sind mit `calc`-Operationen ohne weiteres möglich.

Ein Beispiel:

```
% while-do-Schleife mit Inkrement 1
  \newcounter{i}\setcounter{i}{1}
  \whiledo{\value{i}<4}
    {\thei. Durchgang\
      \stepcounter{i}}
```

Das ergibt folgende Ausgabe:

1. Durchgang
2. Durchgang
3. Durchgang

4.4 Fremdsprachen und Symbole

Zur Erzeugung von fremdsprachigen Symbolen muss nicht das gesamte fremdsprachige package (analog zu `ngerman` oder `german`) eingebunden sein, mit Hilfe der folgenden Eingaben können diese Symbole erzeugt werden.

Eingabe	Ausgabe
<code>\^{o}</code>	ô
<code>\. {o}</code>	ó
<code>\u{o}</code>	ů
<code>\~{o}</code>	õ

Spanisch ist dann halt etwas komplizierter zu schreiben:

```
\' {E}l se\~{n}or est\' {a} bien. \' {E}l est\' {a} aqu\' {i.}
```

Él señor está bien. Él está aquí

Auch Skandinavien wird bestens bedient:¹

Eingabe	Ausgabe
<code>\oe</code>	œ
<code>\OE</code>	Œ
<code>\aa</code>	å
<code>\AA</code>	Å
<code>\ae</code>	æ
<code>\AE</code>	Æ
<code>\o</code>	ø
<code>\O</code>	Ø
<code>\l</code>	ł
<code>\L</code>	Ł

Weitere Symbole, die häufig gebraucht werden sind:

Eingabe	Ausgabe
<code>\S</code>	§
<code>\P</code>	¶
<code>\copyright</code>	©
<code>\pounds</code>	£

4.5 Mathematische Symbole

4.5.1 Grundlegendes

Ganz besonders stark ist LaTeX im Umgang mit mathematischen Formeln; alles, was an Symbolen vorhanden ist, steht zur Verfügung. Wichtig zu wissen ist, dass Formeln im Fliesstextmodus eingegeben werden können, dann stehen sie zwischen einfachen Dollarzeichen, also `$... $`. Wird die Formel als Absatzformel erstellt, steht sie zwischen je zwei Dollarzeichen, also `$$... $$`. Einige sollen hier beispielhaft gezeigt werden:²

Eingabe	Ausgabe
<code>\$x^2\$</code>	x^2
<code>\$x^{2}\$</code>	x^2
<code>\$x^2y\$</code>	x^2y
<code>\$x^{2y}\$</code>	x^{2y}
<code>\$\$\sum\{x_i\}\$\$</code>	$\sum x_i$
<code>\$x_i^2\$</code>	x_i^2
<code>\$x^2_i\$</code>	x_i^2
<code>\$x_{i^2}\$</code>	x_{i^2}

Noch ein paar Beispiele, zuerst als Fliesstextformel, dann als Absatzformeln.

¹Ich kann jetzt Tabellen mit Rahmen versehen.

²weil dieses Thema mich eigentlich nicht interessiert

4. Weiterführende Befehle

Eingabe	Ausgabe
<code>\int_0^\infty</code>	\int_0^∞
<code>\lim_{n \rightarrow \infty}</code>	$\lim_{n \rightarrow \infty}$

`\int_0^\infty` gibt:

$$\int_0^\infty$$

`\lim_{n \rightarrow \infty}` gibt:

$$\lim_{n \rightarrow \infty}$$

Der `_` setzt ein Zeichen tiefer, der `^` setzt eines hoch. Sollen mehr hoch gesetzt werden, setzt man sie in `{}`.

4.5.2 Brüche, Wurzeln

Brüche

Brüche werden mit dem Befehl `\frac{Zähler}{Nenner}` erzeugt; der 1. Parameter ist der Zähler, der 2. ist der Nenner.

Die Eingabe

`\frac{x + y}{z^3 + \frac{1}{a - b}}` ergibt im Druck:

$$\frac{x + y}{z^3 + \frac{1}{a - b}}$$

Mit `\displaystyle` können Formeln „lesbarer“ formatiert werden:

`\frac{x + y}{z^3 + \displaystyle\frac{1}{a - b}}` ergibt im Druck:

$$\frac{x + y}{z^3 + \frac{1}{a - b}}$$

Wurzeln

Wurzeln werden mit dem Befehl `\sqrt{Radikand}` erzeugt.

Der Befehl `\sqrt{x + y}` ergibt $\sqrt{x + y}$.

Der Befehl `\sqrt[3]{x + y}` ergibt $\sqrt[3]{x + y}$.

Die Eingabe `\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{a+b+c+d}}}}}}` liefert das - mathematisch möglicherweise sinnlose, aber ästhetisch umso schönere - Druckbild:

$$\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{a+b+c+d}}}}}}$$

4.6 Mathematische Akzente

Die untenstehende Tabelle zeigt eine kleine Auswahl:

Eingabe	Ausgabe
<code>\$\$\tilde{a}\$\$</code>	\tilde{a}
<code>\$\$\vec{a}\$\$</code>	\vec{a}
<code>\$\$\dot{a}\$\$</code>	\dot{a}
<code>\$\$\overline{x + \overline{y}}\$\$</code>	$\overline{x + \overline{y}}$
<code>\$\$\widetilde{a + b}\$\$</code>	$\widetilde{a + b}$
<code>\$\$\underbrace{u + \dots + u}_{n\text{-mal}}\$\$</code>	$\underbrace{u + \dots + u}_{n\text{-mal}}$

4.6.1 Matrizen

Matrizen funktionieren in der array-Umgebung ähnlich wie Tabellen; statt `\begin{tabular}` und `\end{tabular}` werden die paarweisen Befehle `\begin{array}` und `\end{array}` verwendet, die in ein `\left(` (und ein `\right)` eingeschlossen sein müssen.

Die Eingabe³

```


$$\left(
  \begin{array}{cccc}
    a_{11} & a_{12} & \dots & a_{1n} \\
    a_{21} & a_{22} & \dots & a_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    a_{n1} & a_{n2} & \dots & a_{nn}
  \end{array}
\right)


```

ergibt folgende Ausgabe:

$$\left(\begin{array}{cccc} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{array} \right)$$

und kann natürlich variiert werden:⁴

$$\left\{ \begin{array}{cccc} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{array} \right\}$$

³immer an die Dollarzeichen denken

⁴Wie kann man die `{` und `}` ausgeben?

oder so:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

4.6.2 Gleichungssysteme mit "eqnarray"

Die Gleichungssysteme werden zentriert dargestellt.

Für linksbündige Darstellung von abgesetzten Gleichungen muss `fleqn` (flush left equations) als Option bei der Dokumentenklasse, bzw. bei der Verwendung von AMS- \LaTeX als Paketoption angegeben werden (`\usepackage[fleqn]{amsmath}`).

Gleichungssystem nummeriert

Mit der Umgebung `\begin{eqnarray} - \end{eqnarray}` wird ähnlich der `tabular`-Umgebung das Gleichungssystem dreispaltig, nach dem Gleichheitszeichen ausgerichtet, dargestellt.

Mit `\label{eq:X}` können den einzelnen Gleichungen Referenzen zugewiesen werden, auf welche man sich mit `\ref{eq:X}` beziehen kann. (Siehe dazu auch Kap. 3.15)

Bei Verwendung des Pakets `amsmath` kann die Nummerierung lokal mit `\notag` unterdrückt werden.

```
\begin{eqnarray}
  \sum M \ A \& \& 0 \ \
  \sum M_r \& \& \sum M_l \label{eq:M} \ \
  M_A \& \& M_1+M_2+M_3 \ \
  F_A \cdot l \& \& F_1 \cdot a_1 + F_2 \cdot a_2 + F_3 \cdot a_3 \ \
  \displaystyle F_A \& \&
  \frac{F_1 \cdot a_1 + F_2 \cdot a_2 + F_3 \cdot a_3}{F_A} \ \
  \mathbf{F_A} \& \& \mathbf{5.713kN}
\end{eqnarray}
```

Gleichung `\ref{eq:M}` bedeutet: Die Summe aller rechtsdrehenden Momente ist gleich der Summe aller linksdrehenden Momente.

Erzeugt die folgende Ausgabe:

$$\sum M_A = 0 \quad (4.1)$$

$$\sum M_r = \sum M_l \quad (4.2)$$

$$M_A = M_1 + M_2 + M_3 \quad (4.3)$$

$$F_A \cdot l = F_1 \cdot a_1 + F_2 \cdot a_2 + F_3 \cdot a_3 \quad (4.4)$$

$$F_A = \frac{F_1 \cdot a_1 + F_2 \cdot a_2 + F_3 \cdot a_3}{l} \quad (4.5)$$

$$F_A = \frac{3kN \cdot 0.18m + 5.8kN \cdot 0.305m + 3kN \cdot 0.43m}{0.63m} \quad (4.6)$$

$$\mathbf{F_A} = \mathbf{5.713kN} \quad (4.7)$$

Gleichung 4.2 bedeutet: Die Summe aller rechtsdrehenden Momente ist gleich der Summe aller linksdrehenden Momente.

Gleichungssystem ohne Nummern

Umgebung `\begin{eqnarray*}` - `\end{eqnarray*}`

```

\begin{eqnarray*}
  \sum F_Y &=& 0 \\
  \sum F\downarrow &=& \sum F\uparrow \\
  F_A + F_B &=& F_1 + F_2 + F_3 \\
  F_B &=& F_1 + F_2 + F_3 - F_A \\
  F_B &=& (3+5.8+3-5.713)kN \\
  \mathbf{F_B} &=& \mathbf{6.087kN}
\end{eqnarray*}

```

Erzeugt die folgende Ausgabe:

$$\sum F_Y = 0$$

$$\sum F \downarrow = \sum F \uparrow$$

$$F_A + F_B = F_1 + F_2 + F_3$$

$$F_B = F_1 + F_2 + F_3 - F_A$$

$$F_B = (3 + 5.8 + 3 - 5.713)kN$$

$$\mathbf{F_B} = \mathbf{6.087kN}$$

4.7 Rahmen und Striche

4.7.1 Rahmen

Es gibt drei Arten von Boxen, für deren Einbindung das Package fancybox mit dem Befehl `\usepackage{fancybox}` im Hauptdokument eingebunden werden muss:

LR-Boxen, in denen der Text von links nach rechts ohne Zeilenumbruch gesetzt wird

Parboxen, die einen Absatz mit Zeilenumbruch beinhalten

Rule-Boxen, mit denen Linien und Spalten gezeichnet werden

Folgende Boxen können benutzt werden:

Befehl	Wirkung
<code>\fbox</code>	rahmt den angegebenen Text
<code>\framebox</code>	rahmt den angegebenen Text mit variabler Rahmenbreite
<code>\shadowbox</code>	rahmt den angegebenen Text mit einer Schattenbox
<code>\doublebox</code>	erzeugt einen doppelten Rahmen
<code>\ovalbox</code>	erzeugt einen ovalen Rahmen
<code>\Ovalbox</code>	erzeugt einen fetten ovalen Rahmen
<code>\parbox</code>	erzeugt einen Absatz der mit obigen Kommandos gerahmt werden kann

Beispiele mit LR-Boxen:

Das ist die `\fbox` mit einem Rahmen.

Das ist die `\framebox` mit einem Rahmen.

Das ist die `\shadowbox` mit einem Rahmen.

Das ist die `\doublebox` mit einem Rahmen.

Das ist die `\ovalbox` mit einem Rahmen.

Das ist die `\Ovalbox` mit einem Rahmen.

Beispiele mit Parboxen

Eingabe:

```
Parboxen beinhalten immer einen Zeilenumbruch.\shadowbox{\parbox[c]{6 cm}
{Es kann eine beliebige Breite festgelegt werden; der Umbruch findet dann
innerhalb dieses Absatzes statt. Die Positionsangabe (t für top; b für
bottom; c für center) beeinflusst die Lage des Absatzes bezogen auf die
Grundlinie.}}
```

Ausgabe:

Es kann eine beliebige Breite festgelegt werden; der Umbruch findet dann innerhalb dieses Absatzes statt. Die Positionsangabe (t für top; b für bottom; c für center) beeinflusst die Lage des Absatzes bezogen auf die Grundlinie.


Parboxen beinhalten immer einen Zeilenumbruch.


Beispiele mit Rule-Boxen

Mit folgendem Befehl können Linien und Balken gezeichnet werden:

```
\rule[position]{breite}{höhe}
```

Rechteck 1: 

Rechteck 2: 

Rechteck 3: 

Rechteck 4: 

Kombinationen verschiedener Boxen

Vergleichen Sie diese Box mit dieser Box.

Mit `\raisebox` können Boxen beliebig höher oder niedriger als die Grundlinie ausgegeben werden.

Die `\raisebox{2mm}{Fähre}` `\raisebox{4mm}{zog}` `\raisebox{2mm}{langsam}`
über `\raisebox{-2mm}{das}` Wasser.

Ausgabe:

Die Fähre ^{zog} langsam über das Wasser.

4.8 Nummerierung von Tabellen und Abbildungen

4.8.1 Tabellen nummerieren

Um Tabellen durchnummerieren zu lassen, benutzt man folgende Umgebung:

```

\begin{table}[positionierung der definition]
\begin{tabular}{spaltendefinition}
\end{tabular}
\caption{tabellenunterzeile}
\label{tabellenname}
\end{table}
  
```

Apfel	Birne
Orange	Zitrone

Tabelle 4.1: Das ist eine Beispieltabelle zum Nummerieren

4.8.2 Abbildungen nummerieren

Um Abbildungen durchnummerieren zu lassen, benutzt man folgende Umgebung:

```

\begin{figure}[position]
Abbildung
\caption{Abbildungsunterschrift}
\label{abbildungsname}
\end{figure}
  
```

Beispiel:

Ausgegeben wird die Abbildung dort, wo \LaTeX Platz findet; das kann auch auf der nächsten oder der vorhergehenden Seite sein.

Falls keine Option bei der Position angegeben wird, verwendet \LaTeX als Voreinstellung `[tbp]`. Das bedeutet folgendes (besser ist also die Angabe mit `[h!tbp]`): Wenn man einen schlechten

h	hier, an der Stelle, an der die Abbildung/Tabelle definiert wird
t	oben auf der aktuellen oder nachfolgenden Seite
b	unten auf der aktuellen oder nachfolgenden Seite
p	auf einer eigenen Seite

Tabelle 4.2: Positionsangabe (! zeigt die gewünschte Variante)

Seitenumbruch in Kauf nimmt, kann man die Tabelle dort positionieren, wo sie in der Einbatedatei steht. Man verwendet dazu das Paket `\usepackage{float}` und muss den Befehl `\restylefloat{figure}` absetzen. Die Position wird dann mit `[H]` bestimmt. Analog muss zur Benutzung für Tabellen das Kommando `restylefloat{table}` abgesetzt werden.

4.9 Kopf- und Fusszeilen

Kopf- und Fusszeilen werden über die Befehle
`\pagestyle{seitenstil}` oder `\thispagestyle{seitenstil}`

erzeugt.

Der erste Befehl legt das Seitenlayout entweder im Hauptdokument für das ganze Dokument fest, oder - wird er innerhalb eines Dokumentes verwendet - für die aktuelle und die folgenden Seiten. Mit dem zweiten Befehl wird nur die aktuelle Seite beeinflusst. Wird keiner der beiden Befehle benutzt, gelangt die Seitenzahl standardmässig in den Seitenfuss, und zwar zentriert; der Seitenkopf bleibt dann leer.

Für den Seitenstil gelten folgende Parameter:

Parameter	Wirkung
<code>empty</code>	Seitenkopf und -fuss bleiben leer
<code>plain</code>	Kopfzeile bleibt leer; Fusszeile enthält Seitennummerierung (default)
<code>headings</code>	Kopfzeile enthält die aktuelle Überschrift und die Seitenzahl; Fusszeile ist leer
<code>myheadings</code>	Kopfzeile wird vom Benutzer festgelegt

Abhängig von der verwendeten Dokumentenklasse werden die Überschriften automatisch in die Kopfzeile übernommen. Bei zweiseitigen Dokumenten der Klasse `article` steht auf der linken Seite als Kopfzeile der Titel der `section`, auf der rechten Seite als Kopfzeile der Titel der `subsection`.

Wird der Parameter `myheading` verwendet muss mit einem weiteren Befehl die Kopfzeile definiert werden:

```
\thispagestyle{myheadings}
\markboth{Kopf linke Seite}{Kopf rechte Seite}
zum Beispiel:
\thispagestyle{myheadings}
\markboth{testlinks}{testrechts}
```

Kapitel 5

Hilfreiche Befehle

5.1 Integration von PS-Grafiken

Notwendig für die Integration von Postscript-Grafiken ist das Paket, das folgendermassen aufgerufen wird:

```
\usepackage[dvips]{graphicx}
```

5.2 Verdrehen von Objekten

Notwendig für die Integration von Postscript-Grafiken ist das Paket, das folgendermassen aufgerufen wird:

```
\usepackage[dvips]{rotating}
```

Beides scheint sich mit `\usepackage{thumbpdf}` zu beissen.

5.3 Mehrspaltendruck

Mehrspaltiger Text kann im Kommando `\documentclass` als Option festgelegt werden. Wenn das nicht festgelegt worden ist, braucht es für Mehrspaltendruck das Paket, das folgendermassen aufgerufen wird:

```
\usepackage{multicol}
```

Im Dokument kann das Paket dann benutzt werden durch:

```
\begin{multicols}{spaltenanzahl}[titel][abstand]  
...  
...  
\end{multicols}
```

Über `\setlength{\columnsep}{breite}` legt man den Spaltenabstand fest und über `\setlength{\columnseprule}{strichbreite}` wird die Trennungstrichbreite festgelegt.

Ein Beispiel:

Ein Goldschmied aus Mainz erfand Mitte des 15. Jahrhunderts in Europa den Buchdruck mit beweglichen Lettern. Johann Gensfleisch Gutenberg benutzte eine alte Weinpresse zum Drucken der Bücher. Sie war die erste Druckerpresse der Welt, mit der eine neue Symbolwelt geschaffen wurde und die somit das Informationszeitalter einleitete.

Vom Buchdruck zum Personal Computer

Mit den Büchern erhielten die Menschen Informationen, die bis dahin für viele unzugänglich waren. Die Geschäftswelt wandelte sich infolge der steigenden Nutzung von Verträgen, Urkunden, Schuldscheinen und Landkarten in eine Welt aus bedrucktem Papier. Die Form des ge-

druckten Buches erzeugte eine neue Methode Inhalte zu organisieren und förderte das organisierte Denken. Auch Samuel Morse von der New York University ahnte kaum, welche Auswirkungen seine Erfindung auf die Entwicklung der Kommunikation hatte. Zum ersten Mal konn-

te der Mensch mit Hilfe der Elektrizität Informationen zwischen weit entfernten Orten übermitteln und austauschen. Der Telegraphenapparat war ein neutrales und unpersönliches Vermittlungsinstrument.

Der elektromagnetische Telegraph brachte die Nachrichtenindustrie hervor. Der Telegraph beseitigte den Raum als unausweichliche Schranke der Informationsbewegung. Er koppelte die Information zum ersten Mal vom Verkehrswesen ab. Bereits im 18. Jahrhundert entwickelte der begabte englische Mathematiker Charles Babbage (1792 1871) eine me-

chanisch gesteuerte Rechenmaschine. Die Idee kam ihm, als er, knapp zwanzigjährig, damit beschäftigt war, fehlerhafte Logarithmentafeln zu korrigieren. Seine Konstruktionen und Ideen stiegen aber überall auf technisches Unverständnis. So starb er, bitter enttäuscht vom Leben und den Ergebnissen seines Schaffens, ohne dass sein Lebenswerk vollendet wurde.

Erst 100 Jahre später waren der Mensch und die Technik in der Lage, seine bahnbrechenden Ideen zu verstehen und zu verwirklichen. Unabhängig voneinander entwickelten in Deutschland Konrad Zuse (1934) und in Amerika Howard Aiken (1939) Rechenautomaten.

Im Beispiel war der Text mit `\small` verkleinert worden; ab hier ist er mit Hilfe des Befehls `\normalsize` wieder in normaler Grösse gesetzt.

5.4 Noch viel mehr Schriften

L^AT_EX bietet noch viel mehr Schriften; es ist wie Winword eine eigentliche Schriftenmustersentrale. Diese Schriften werden wie folgt ausgewählt:

```

\fontencoding{...}
\fontfamily{...}
\fontseries{...}
\fontshape{...}
\fontsize{...}{...}
\selectfont{...}

```

Was bedeutet das?

fontencoding dient zur Angabe des Zeichensatzes

fontfamily wählt eine Schriftfamilie aus

fontseries gibt die Laufweite und Stärke der Schrift an

fontshape bestimmt die aufrechte oder kursive Form eines Zeichensatzes

fontsize definiert die Grösse einer Schrift und den Zeilenabstand

Das Beispiel: Der Befehl

`\fontfamily{cmdh}\fontseries{m}\fontshape{n}\fontsize{14.4}{17pt}\selectfont`
 wählt die Schrift **Computer Modern Dunhill** , danach wird mit folgendem Kommando

```
\fontfamily{\familydefault}
\fontseries{\seriesdefault}
\fontshape{\shapedefault}
\normalsize
\selectfont
```

wieder auf normale Schrift umgestellt.

5.4.1 Alte deutsche Schriften

Ganz schön sind die alten deutschen Schriften:

Old german Fonts: Package yfonts		
Family	Shape	Beispiel
Fraktur	<code>\frakfamily</code>	Frakturschrift: Dem Vogt eine Gasse !
Schwabinger	<code>\swabfamily</code>	Schwabinger: Dem Vogt eine Gasse !
Goth	<code>\gothfamily</code>	Goth: Dem Vogt eine Gasse !

5.4.2 Initialen

Für initialen muss das Packet `yfonts` geladen werden, für Deutsch am Besten so:
`\usepackage[varumlaut]{yfonts}` (dies wird in `bbb.sty` bereits gemacht). Nun können Initialen folgendermassen erzeugt werden:

```
\fraklines
\yinipar{\color{blue}B}lah, blahblah...
```



ns wird in alten Mären wonders viel geseit von Helden lobebären und groser Arebeit vn Freuden Hochgeziten, von Weinen und von Klagen von kuener Recken striten muget ir nu hören sagen. ns wird in alten Mären wonders viel geseit von Helden lobebären und groser Arebeit vn Freuden Hochgeziten, von Weinen und von Klagen von kuener Recken striten muget ir nu hören sagen. ns wird in alten Mären wonders viel geseit von Helden lobebären und groser Arebeit vn Freuden Hochgeziten, von Weinen und von Klagen von kuener Recken striten muget ir nu hören sagen.

5.5 Nützliche Symbole










In BBB.STY definiert, wenn es mit der Option `symbolfonts` aufgerufen wird. Allerdings sind nicht alle Packages in Debian enthalten.

^	<code>\booland</code>	∨	<code>\boolor</code>
---	-----------------------	---	----------------------

In DINGBAT.STY definiert:



Zusätzlich noch `\largepencil` () der nicht in die Tabelle passt.

	<code>\leftpointright</code>		<code>\rightpointleft</code>
	<code>\rightpointright</code>		
	<code>\leftthumbsdown</code>		<code>\rightthumbsdown</code>
	<code>\leftthumbsup</code>		<code>\rightthumbsup</code>
	<code>\smallpencil</code>		<code>\eye</code>
↵	<code>\carriagereturn</code>	✓	<code>\checkmark</code>

In WASYSYM.STY definiert:

~	<code>\AC</code>	≈	<code>\HF</code>	⚡	<code>\lightning</code>	✓	<code>\checked</code>
♂	<code>\male</code>	♀	<code>\female</code>	🕒	<code>\clock</code>	☞	<code>\pointer</code>
☺	<code>\smiley</code>	☹	<code>\frownie</code>	☹	<code>\blacksmiley</code>		
🔔	<code>\bell</code>	☎	<code>\phone</code>	‰	<code>\permil</code>	↯	<code>\varangle</code>
☑	<code>\CheckedBox</code>	☐	<code>\Square</code>	☒	<code>\XBox</code>		

In MARVOSYM.STY definiert:

€	<code>\EUR</code>	€	<code>\EURcr</code>	€	<code>\EURhv</code>	@	<code>\Ecommerce</code>
✉	<code>\Email</code>	✉	<code>\Letter</code>	📞	<code>\Mobilefone</code>	☎	<code>\Telefon</code>
☢	<code>\Radioactivity</code>	🛑	<code>\Stopsign</code>				
🖱	<code>\ComputerMouse</code>	⌨	<code>\Keyboard</code>	🖨	<code>\Printer</code>	📠	<code>\Faxmachine</code>
☑	<code>\Checkedbox</code>	☒	<code>\Crossedbox</code>	☕	<code>\Coffeecup</code>	⚽	<code>\Football</code>
✂	<code>\Cutright</code>	---	<code>\Cutline</code>	---	<code>\Kutline</code>	✂	<code>\Cutleft</code>
✍	<code>\Writinghand</code>	ℹ	<code>\Info</code>	✂	<code>\Rightscissors</code>	✂	<code>\Leftscissors</code>
🦇	<code>\Bat</code>	🌸	<code>\Bouquet</code>	🐦	<code>\MartinVogel</code>	👩	<code>\Womanface</code>
🌐	<code>\Mundus</code>	⊕	<code>\CircledA</code>	♥	<code>\Heart</code>	☯	<code>\Yinyang</code>

Weiter könnte man mal IFSYM ansehen. Damit kann man Uhren mit einer vorgegebenen Zeit und Signalverläufe darstellen.

5.6 Setzen von Listings

Um Listings zu setzen, wird in BBB.STY das Package listings importiert. Einigermassen sinnvolle Werte wurden gesetzt, doch LISTINGS kann noch viel mehr. Wie immer sei der Interessierte auf die Dokumentation verwiesen.

Hier ein paar Beispiele:

```
\begin{lstlisting}[language=Java]
package msc;
import robocode.*;
//import java.awt.Color;
\end{lstlisting}
```

```
1 package msc;
2 import robocode.*;
3 //import java.awt.Color;
4
5 /**
6  * StupidZonk – a robot by (your name here)
7  */
8 public class StupidZonk extends Robot
9 {
10     /**
11     * run: StupidZonk's default behavior
12     */
13     public void run() {
14         // After trying out your robot, try uncommenting the
15         // import at the top,
16         // and the next line:
17         //setColors(Color.red, Color.blue, Color.green);
18         while(true) {
19             // Replace the next 4 lines with any behavior you
20             // would like
21             ahead(100);
22             turnGunRight(360);
23             back(100);
24             turnGunRight(360);
25         }
26     }
27     /**
28     * onScannedRobot: What to do when you see another robot
29     */
30     public void onScannedRobot(ScannedRobotEvent e) {
31         fire(1);
32     }
33     /**
34     * onHitByBullet: What to do when you're hit by a bullet
35     */
36     public void onHitByBullet(HitByBulletEvent e) {
37         turnLeft(90 - e.getBearing());
38     }
}
```

5. Hilfreiche Befehle

39

40 }

Oder aus BBB.STY:

```

\begin{lstlisting}[language=TeX]
% Document History
\newenvironment{scripthistory}{
...
\end{lstlisting}
1 % Document History
2 \newenvironment{scripthistory}{
3     \newcommand{\historyitem}[4]{##1 & ##2 & ##3 & ##4 \\}
4     \begin{tabular*}{\textwidth}{@{\extracolsep{\fill}}|c|r|c|p{9cm}|}
5     \hline \textbf{Datum} & \textbf{Version} & \textbf{Autor} & \textbf{
        Bemerkungen} \\ \hline
6 }{
7 \hline
8 \end{tabular*}
9 }
  
```

5.7 Zitate

Eine Möglichkeit, seine Texterzeugnisse geistreicher erscheinen zu lassen ist es, zu Beginn eines Kapitels ein mehr oder weniger passenden Spruch zu platzieren.

Sprüche gibt es unter <http://www.quotationspage.com/> (und weiteren). gesetzt werden können sie in L^AT_EX folgendermassen:

```

\begin{quotation}
\noindent The hardest thing is to go to sleep at night,
when there are so many urgent things needing to be done.
A huge gap exists between what we know is possible with today's machines
and what we have so far been able to finish.

-- Donald Knuth, inventor of \LaTeX\
\end{quotation}
  
```

The hardest thing is to go to sleep at night, when there are so many urgent things needing to be done. A huge gap exists between what we know is possible with today's machines and what we have so far been able to finish.

– Donald Knuth, inventor of L^AT_EX

Kapitel 6

PDF Spezialitäten

6.1 Dateien in PDF einbetten

Wird `bbb.sty` mit der Option `attachfile` geladen, dann wird das Package `attachfile.sty` eingebunden. Dieses Package erlaubt das Einbinden von irgendwelchen Dateien in pdf-Dokumente. Die Möglichkeiten sind vielfältig: Texte/Programmcodes, Animationen, Webseiten, Flash, ... Alles möglich !

In `BBB.STY` gibt es die Funktion `\attachhere[]{}{}`, die auch in PS-Dateien einen einigermaßen vernünftigen Output gibt.

Beispielsweise das HelloWorld in Java.

Ganz einfach:

[HelloWorld.Java](#)

```
\attachhere{images/HelloWorld.java}{HelloWorld.Java}
```

Oder etwas luxuriöser: [HelloWorld.Java](#)

```
\attachhere[author={Michael Schneider},subject={HelloWorld in Java},
description={Das einfachste Javaprogramm}]
{images/HelloWorld.java}{HelloWorld.Java}
```

Webseiten: [si.html](#)

```
\attachhere[subject={SI-Einheiten}]{images/si.html}{si.html}
```

Oder auch Filme: [random.mpg](#) [random.mpg](#)

```
\attachhere[description={MPEG Film}]{images/random.mpg}{random.mpg}
```

Sound: [manonmoon.mp3](#)

```
\attachhere[subject={Man on Moon},
description={Die berühmten Worte des Mannes im Mond}]
{images/manonmoon.mp3}{manonmoon.mp3}
```

6. PDF Spezialitäten

Flash-Tools: [froggie.swf](#)

```
\attachhere[subject={Frogger lebt!},description={Wer kennt das noch.  
Natürlich könnte man auch  
Lernprogramme in Flash einbetten !}]  
{images/froggie.swf}{froggie.swf}
```

Am besten gehen andere PDF-Dokumente: [unix-ref.pdf](#)

```
\attachhere[description={Unix reference}]{images/unix-ref.pdf}{unix-ref.pdf}
```

Kapitel 7

Indices

Wird BBB.STY verwendet, kann ein Index und ein Glossar erstellt werden. Wird das MAKEFILE benutzt, so genügt die Konfiguration:

```
WITH_INDEX = Yes
WITH_GLOSSARY = Yes
```

7.1 Index

Um ein Index zu erstellen, wird hinter dem zu indizierenden Wort `\index{WORT}` angehängt. Dabei gibt es für `WORT` noch einige Goodies:

Funktion	Beispiel	Beschreibung
Indexeintrag	<code>\index{alpha}</code>	<code>alpha</code> wird in den Index eingetragen.
Untereintrag	<code>\index{Indices!Index}</code>	<code>Index</code> wird in den Index als Untereintrag zu <code>Indices</code> eingetragen.
Verweis	<code>\index{Wortverzeichnis see Glossar}</code>	<code>Wortverzeichnis</code> wird mit einem Verweis zu <code>Glossar</code> in den Index eingetragen.
Umsortierung	<code>\index{zweiundvierzig@42}</code>	<code>42</code> wird im Index eingeordnet wie <code>zweiundvierzig</code> .

Die Indexgenerierung läuft so ab, dass die Quelldatei zuerst durch \LaTeX gelassen werden muss. daraus entsteht die Datei `QUELLDATEI.IDX`. Danach muss der Index erschaffen werden. Unter Linux geschieht dies durch `makeindex quelldatei.idx` oder im BBB Umfeld mittels `makeindex -g -s include/bbbind.idx`, welches einen schöneren, deutsch geordneten Index erstellt. Wird diese Datei nun nochmals durch \LaTeX hindurchgelassen, dann gibt es am Schluss einen solch imposanten Index, wie ihn dieses Dokument besitzt.

7.2 Glossar

Ein Glossar, bzw. eine Begriffserklärung funktioniert ähnlich wie ein Index. Im Text muss `\nomenclature{Begriff}{Erklärung}` verwendet werden.

In der Erklärung können lokal mit drei Makros Referenzen eingefügt werden:

Makro	Auswirkung
<code>\nomrefeq</code>	Gleichungsnummer der letzten Gleichung
<code>\nomrefeqpage</code>	Gleichungsnummer und Seitennummer der letzten Gleichung
<code>\nomrefpage</code>	Seitennummer

Der Code

```
\nomenclature{ $\sum M \ A$ }{Summe aller Momente um den Lagerpunkt A}
```

ergibt die folgende Ausgabe:

$\sum M \ A$: Summe aller Momente um den Lagerpunkt A, siehe Gleichung (4.2), Seite 27

Der Index wird nach dem ersten \LaTeX run ganz ähnlich erzeugt:

```
makeindex -s nomencl.ist -o quelle.gls quelle.glo.
```

Kapitel 8

Grafiken

8.1 Grafikformate

Mit `BBB.STY` sind die Grafiken folgendermassen Konfiguriert:

- Für die PostScript und DVI-Variante verwendet er Encapsulated PostScript (EPS) Dateien.
- Für das PDF verwendet er die folgende Reihenfolge:
 1. PDF Grafiken (Vorteil: Sind Vektororientiert und werden so beim Zoom nicht Pixelig.
 2. PNG
 3. JPG
 4. GIF

Jede Grafik muss dementsprechend *zwei* mal vorhanden sein: Eine für das PDF und eine für die PostScript Variante.

8.2 Grafiken als Figures/Floats

Im Normalfall setzt man Grafiken als *Floats*. Floats sind bewegliche Bereiche, die \LaTeX so positioniert, wie es halt passt und eine Falle macht. In einem solchen Float können nicht nur Bilder, sondern auch Tabellen (diese allerdings nicht als `figure` sondern als `table`) und andere Texte eingepasst werden. Ein Beispiel ist die Abbildung [8.1](#) auf der nächsten Seite.

Der Quellcode sieht folgendermassen aus:

```
1 \begin{ figure }[h!tb]
2 \begin{ center }
3 \includegraphics [ width=50mm]{ images/m sword }
4 \end{ center }
5 \caption{ WYSIWYG in Aktion } \label{ fig : m sword }
6 \end{ figure }
```

Beschreibung:

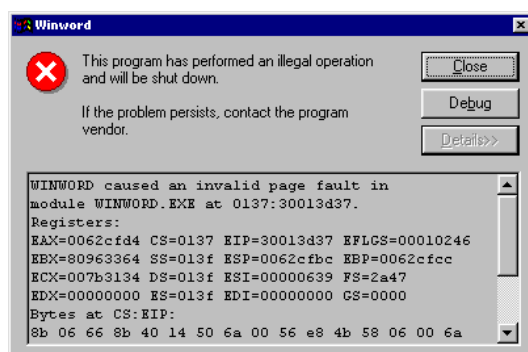


Abbildung 8.1: WYSIWYG in Aktion

Zeile 1: Die eckige Klammer sagt aus: Versuch es sehr fest, das Ding hier zu platzieren (**h!**).
Sonst mach es irgendwo oben oder unten hin.

Zeile 2 und 4: Das Ganze horizontal zentriert.

Zeile 3: Hier wird die Grafik eingebunden. Achtung, *ohne* Dateiendung !

Zeile 5: Die Bildunterschrift und ein Label, um darauf referenzieren zu können (siehe Kapitel 3.15 auf Seite 17).

8.3 Grafiken im Text

Mit PICINS können texte um Grafiken herumfließen. Der Syntax ist der folgende:

```
\parpic(w,h)(x-o,y-o)[opt][pos]{pict}
```

Wobei:

w, h: Breite und Höhe der Grafik. Optional wenn \LaTeX diese selbst herausfinden kann.

x-o, y-o: Offset zum Verschieben des Bildes.

opt: Optionale Platzierung und Aussehen:

l: Links des Absatzes.

r: Rechts des Absatzes.

d: Gestrichelte Linie.

f: Normale Linie.

o: Abgerundetes Rechteck-


s: Schattierte Box.


x: 3D-Box.


pos: Optionale Position des Bildes innerhalb des ihm zur Verfügung stehenden Platzes: l, r, t, b.


pict: Irgendein beliebiges Latex-Konstrukt.


Beispiele für Picins


- 

Dies wurde mit dem mächtigen Packet PICINS erzeugt. Der Text fließt schön um die Grafik, was mit Floats nicht möglich ist. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Tralla tralla trallalla! Tralla tralla trallalla! Tralla tralla trallalla!
- 

Dies wurde mit dem mächtigen Packet PICINS erzeugt. Der Text fließt schön um die Grafik, was mit Floats nicht möglich ist. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Tralla tralla trallalla! Tralla tralla trallalla!
- 

Dies wurde mit dem mächtigen Packet PICINS erzeugt. Der Text fließt schön um die Grafik, was mit Floats nicht möglich ist. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Tralla tralla trallalla! Tralla tralla trallalla! Tralla tralla trallalla!
- 

Dies wurde mit dem mächtigen Packet PICINS erzeugt. Der Text fließt schön um die Grafik, was mit Floats nicht möglich ist. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Tralla tralla trallalla! Tralla tralla trallalla!
- 

Dies wurde mit dem mächtigen Packet PICINS erzeugt. Der Text fließt schön um die Grafik, was mit Floats nicht möglich ist. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Tralla tralla trallalla! Tralla tralla trallalla! Tralla tralla trallalla!
- 

Dies wurde mit dem mächtigen Packet PICINS erzeugt. Der Text fließt schön um die Grafik, was mit Floats nicht möglich ist. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Das hinterlässt einen fließenden Eindruck Blahblah. Tralla tralla trallalla! Tralla tralla trallalla!

Code:

```

1 \begin{multicols}{2}[Beispiele für Picins]
2 \begin{itemize}
3   \item \parpic{\includegraphics [width=20mm]{images/latex}} Dies wurde
      mit dem
4     ...
5   \item \parpic (22mm,22mm) [dr]{\includegraphics [width=20mm]{images/latex
      }} Dies wurde mit dem
6     ...
7   \item \parpic (22mm,22mm) [fl]{\includegraphics [width=20mm]{images/latex
      }} Dies wurde mit dem
8     ...
9   \item \parpic (30mm,30mm) [or]{\includegraphics [width=20mm]{images/latex
      }} Dies wurde mit dem
10    ...
11  \item \parpic (22mm,22mm) [sr]{\includegraphics [width=20mm]{images/latex
      }} Dies wurde mit dem
12    ...
13  \item \parpic (22mm,22mm) [xr]{\includegraphics [width=20mm]{images/latex
      }} Dies wurde mit dem
14    ...
15 \end{itemize}
16 \end{multicols}
  
```

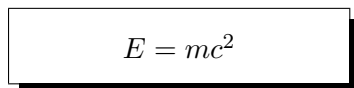


Abbildung 8.2: Die Einsteinsche Formel

Man kann auch andere Konstrukte (wie zum Beispiel Formeln oder Symbole) als Bilder in solche Umgebungen einfügen. Das Kommando `\piccaption{Beschreibung}` nimmt dabei die Beschreibung für die nächste Parpicumgebung auf. Der Nachteil ist, dass die Nummern mit den anderen Figures nicht in Sequenz stehen, da \LaTeX diese Parpics nicht herumschieben darf.

Dieses Beispiel wurde folgendermassen erzeugt:

```

\piccaption{Die Einsteinsche Formel}
\parpic(45mm,10mm) [s]{ $E=mc^2$ } Man kann auch andere...
  
```

8.4 Grafiken mit PGF

8.4.1 Gebrauch und Installation

Zuerst ein grosser Dank an Ernst, der mich mit seinen schönen PSTricks grafiken angestachelt hat, was zu suchen das für PDF und PS gleichermassen funktioniert.

Es sind nicht alle libraries für pgf installiert. Wenn jemand mehr braucht, kann man die gerne einklinken.

Linux

Um PGF zu benutzen, muss das ganze Package installiert sein und die Option `pgf` muss beim `bbb` Package angegeben werden. Damit es funktioniert braucht es das Paket `XKEYVAL` in der

Version 2.5 oder höher installiert sein (was unter Debian standardmässig nicht der Fall ist).
 Bei Fehlermeldungen wie unten gezeigt, ist ein Update angebracht:

```
! Package xkeyval Error: 'moretexcs' undefined in families 'lst'.
! LaTeX Error: Missing \begin{document}.
```

Windows

Ernst Mösching:

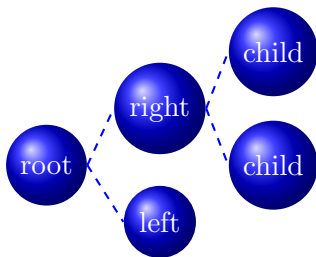
Notwendige Einstellungen für Windows um ein Funktionsdiagramm wie in Kapitel 8.4.2 auf Seite 52 ff. zu erstellen:

- Gnuplot installieren
- wgnuplot.exe kopieren nach gnuplot.exe
- gnuplot-Pfad eintragen in Arbeitsplatz/ Eigenschaften/ Erweitert/ Umgebungsvariablen/ Systemvariablen/ Path
 Beispiel: c:\programme\gnuplot\bin\
- Für Kompilation bei pdflatex Schalter `--enable-write18` setzen

8.4.2 Beispiele

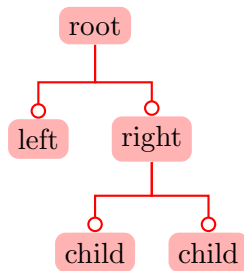
Bäume

Bäume können auf die verschiedensten Arten aufgezogen werden:



```

1 \begin{tikzpicture}[parent anchor=east,child anchor=west,grow=east]
2 \tikzstyle{every node}=[ball color=blue,circle,text=white]
3 \tikzstyle{edge from parent}=[draw,dashed,thick,blue]
4 \node {root}
5 child {node {left}}
6 child {node {right}}
7 child {node {child}}
8 child {node {child}}
9 };
10 \end{tikzpicture}
  
```



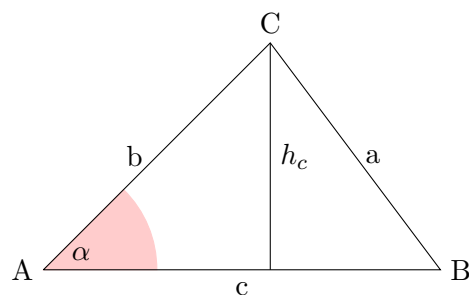
```

1 \begin{tikzpicture}[edge from parent fork down]
2 \tikzstyle{every node}=[fill=red!30,rounded corners]
3 \tikzstyle{edge from parent}=[red,-o,thick,draw]
4 \node {root}
5 child {node {left}}
6 child {node {right}}
7 child {node {child}}
8 child {node {child}}
9 };
10 \end{tikzpicture}

```

Schiefwinkliges Dreieck

Beispiel von Ernst Mösching:



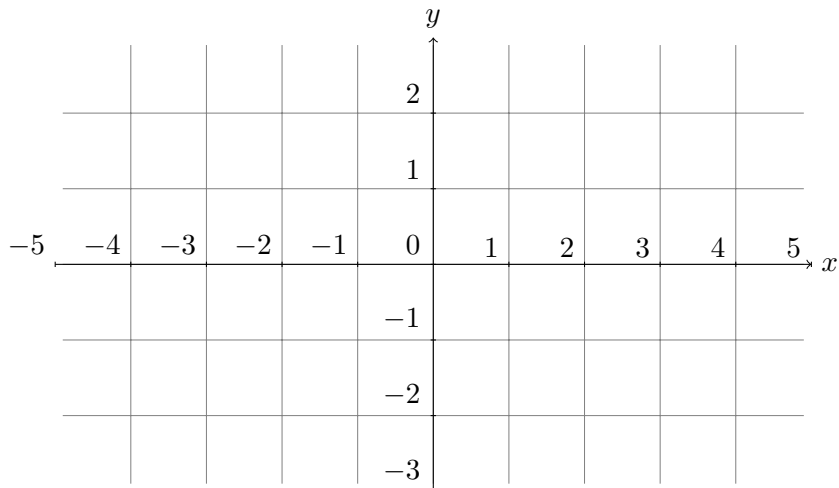
```

1 \begin{tikzpicture}[scale=1.5]
2 \filldraw[fill=red!20,draw=red!20] (0,0) --(1,0) arc (0:45:1cm) -- cycle
3 ;
4 \draw
5 (0,0) coordinate(A) node[anchor=east]{A} -- node[left=1.5pt]{b}
6 (2,2) coordinate(C) node[anchor=south]{C} -- node[right]{a}
7 (3.5,0) coordinate(B) node[anchor=west]{B} -- node[below=1pt]{c}
8 (0,0);
9 \draw (C) -- node[right]{\$h_c\$} (2,0);
10 \draw (A) node[right=0.5cm,above]{\$alpha\$};
11 \end{tikzpicture}

```

Diagramme

Einfache Koordinatensysteme:

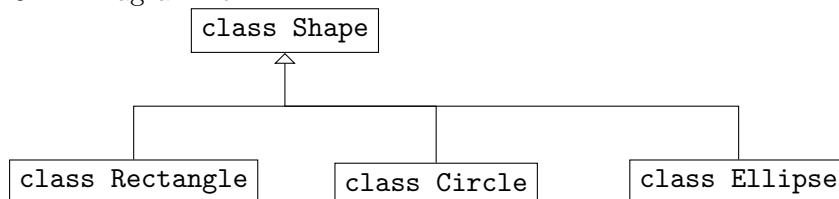


```

1 \begin{tikzpicture}
2 \draw[step=1.0cm,gray,very thin] (-4.9,-2.9) grid (4.9,2.9);
3 \draw[->] (-5,0) -- (5,0) node[right] {$x$};
4 \draw[->] (0,-3) -- (0,3) node[above] {$y$};
5 \foreach \x in {-5,...,-1,1,2,...,5}
6   \draw (\x cm,1pt) -- (\x cm,-1pt) node[anchor=south east] {$\x$};
7 \foreach \y in {-3,...,2}
8   \draw (1pt,\y cm) -- (-1pt,\y cm) node[anchor=south east] {$\y$};
9 \end{tikzpicture}

```

UML-Diagramme:



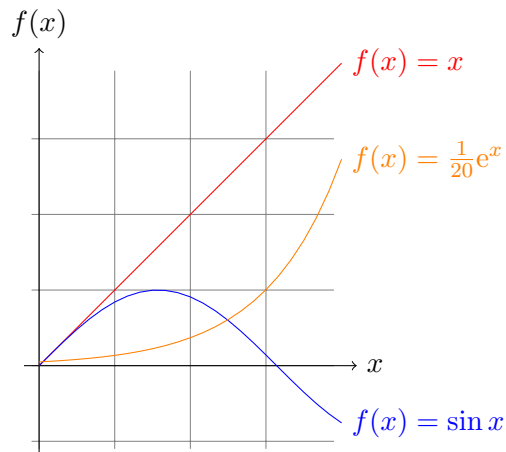
```

1 \begin{tikzpicture}
2 \tikzstyle{every text node part}=[font=\ttfamily]
3 \node (shape) at (0,2) [draw] {class Shape};
4 \node (rect) at (-2,0) [draw] {class Rectangle};
5 \node (circle) at (2,0) [draw] {class Circle};
6 \node (ellipse) at (6,0) [draw] {class Ellipse};
7 \draw (circle.north) |- (0,1);
8 \draw (ellipse.north) |- (0,1);
9 \draw[-open triangle 90] (rect.north) |- (0,1) -| (shape.south);
10 \end{tikzpicture}

```

8. Grafiken

Für folgendes Beispiel muss GNUPLOT installiert sein und beim Kompilieren muss Latex die Option `--shell-escape` (in Linux), bzw. `--enable-write18` (unter Windows) mitgegeben werden:

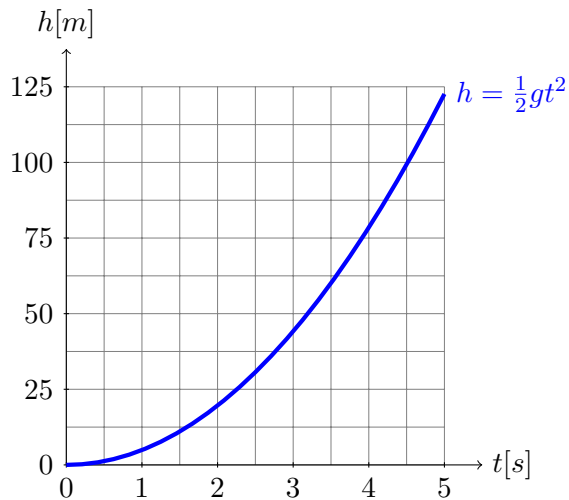


```

1 \begin{tikzpicture}[domain=0:4]
2 \draw[very thin,color=gray] (-0.1,-1.1) grid (3.9,3.9);
3 \draw[->] (-0.2,0) -- (4.2,0) node[right] {$x$};
4 \draw[->] (0,-1.2) -- (0,4.2) node[above] {$f(x)$};
5 \draw[color=red] plot[id=x] function{x} node[right] {$f(x) = x$};
6 \draw[color=blue] plot[id=sin] function{sin(x)} node[right] {$f(x) = \sin
  x$};
7 \draw[color=orange] plot[id=exp] function{0.05*exp(x)} node[right] {$f(x)
  = \frac{1}{20} \mathrm{e}^x$};
8 \end{tikzpicture}
  
```

8. Grafiken

Skalierbare Achsen:

Der freie Fall


```

\begin{tikzpicture}[scale=1,domain=0:5]
  \draw[step=0.5cm,style=help lines] (0,0) grid (5,5);
  \draw[->] (0,0) -- (5.5,0) node[right]{$t[s]$};
  \draw[->] (0,0) -- (0,5.5) node[above]{$h[m]$};
  \foreach \x in {0,1,...,5}
    \draw (\x cm,1pt) -- (\x cm,-1pt) node[below] {$\x$};
  \foreach \y/\ytext in {0/0,1/25,2/50,3/75,4/100,5/125}
    \draw (1pt,\y cm) -- (-1pt,\y cm) node[left] {$\ytext$};
  \draw[color=blue, ultra thick] plot[id=h] function{(.5*9.81*x**2)/25}
    node[right]{$h=\frac{1}{2}gt^2$};
\end{tikzpicture}

```

Anhang A

BBB-Standards

Dies ist ein RFC für verschiedene Standards an der BBB. Die erleichtert sowohl den Unterhalt, die Übernahme, als auch die Ergänzung von Scripts. Auch für die Lernenden ist eine konsistente Schreibweise hilfreich.

A.1 SVN Verzeichnisstruktur

Das oberste Verzeichnis eines SVN Repositories beinhaltet die folgenden Verzeichnisse:

BRANCHES: Für verzweigte Entwicklungspfade.

TAGS: Zur Speicherung eines Zustandes zu einem bestimmten Zeitpunkt.

TRUNK: Das eigentliche Arbeitsverzeichnis (und oftmals auch das Einzige mit Inhalt).

TRUNK/ beinhaltet folgende Verzeichnisse:

DIVERSES/: Beinhaltet den methodisch-didaktischen Beschrieb, Klassenlernpläne und Modulbeschriebe.

DIVERSES/PRAESENTATIONEN: Präsentationen.

LA: Lern- und Arbeitsaufträge.

SCRIPT: Verzeichnisse für Scripts, jeweils in ihren eigenen Unterverzeichnissen.

SCRIPT/BBB: Von der BBB erstellte Scripts.

TESTS: Prüfungen zu diesem Thema/Modul.

A.2 L^AT_EX Verzeichnisstruktur

Wird ein Script in L^AT_EX geschrieben, sollte der Aufbau folgendermassen sein:

/: (Grundverzeichnis) Beinhaltet *nur* die Hauptdatei im .TEX Format, das MAKEFILE und Konfigurationsdateien (sowie eventuell BBB.STY. Es gibt *keinen* Grund, temporäre Dateien einzuchecken.

BUILD: Die PDF Datei. Diese wird auch in Moodle referenziert und beinhaltet einen stabilen Stand.

SEC: Die einzelnen Kapitel.

IMAGES: Bilder im .EPS und jeweils ein zweites, PDF-gängiges Format sowie weitere Medien.

INCLUDE: Das Indextemplate und allgemeine dateien wie das BBB- und die Prozesslogos.

HTML: Die HTML-Ausgabe des Scripts.

A.3 Schreibweisen

Was	Macro	Beispiel
Geschriebener Text	<code>\bbdtype {}</code>	Tippen Sie <code>logout</code> in die Konsole. Tippen Sie <code>\bbdtype {logout}</code> in die Konsole.
Menüs	<code>\bbbmenu {}</code>	Wählen Sie Datei \Rightarrow Speichern. Wählen Sie <code>\bbbmenu {Datei \$\rightarrow\$ Speichern}</code> .
Dateinamen	<code>\bbbfile {}</code>	Öffnen Sie <code>LATEX.TEX</code> . Öffnen Sie <code>\bbbfile {LaTeX.tex}</code> .

A.4 Allgemeines

- Der Lernende wird mit *Sie* angesprochen.
- Es gilt die neue deutsche Rechtschreibung.

Anhang B

BBB-Styles

Hier stehen Kurzbeschreibungen und Gebrauchsanweisungen für BBB-Styles. Damit soll auch der Erstanwender rasch ein BBB-Style einsetzen können. Für weitergehende Anweisungen und Erklärungen sei wie immer, auf die einschlägige Literatur verwiesen.

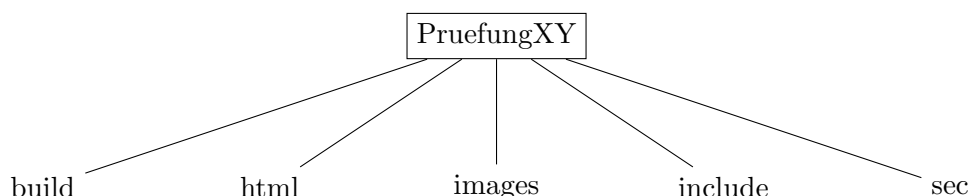
B.1 Prüfungen erstellen mit `bbb-new-test.sty`

Das von Michael Schneider erstellte `bbb-new-test.sty` verwendet die `exam`-Klasse von [Philip Hirschhorn](#).

Im BBB-SVN, z.B. unter [SVN-Maschinenbau](#) sind viele Prüfungen abgelegt, die als Vorlage oder Demo verwendet werden können. Nachdem dann persönliche Standards für die eigenen \LaTeX -Prüfungen einmal feststehen, kann jeweils für eine neue Prüfung lediglich eine gesamte bestehende \LaTeX -Verzeichnisstruktur kopiert und abgeändert werden.

Zur Erstellung der ersten \LaTeX -Prüfung mit `bbb-new-test.sty` empfiehlt sich folgendes Vorgehen:

1. Lokales Verzeichnis mit dem Namen der Prüfung erstellen, z.B. `PruefungXY`.
2. Darin BBB-Latexstruktur gemäss Kap. [A.2](#) erstellen:



3. Kopieren der Dateien `bbbaden.eps` und `bbbaden.png` aus dem Verzeichnis `include` von

<https://svn.bbbaden.ch/websvn/informatik/listing.php?repname=LaTeX-Kurs&path=%2Ftrunk%2FBBB-Style%2Ftest%2FNewClass%2Fexample%2F&rev=0&sc=0>

in das lokale `include`-Verzeichnis.

4. Mit obigem Link die Dateien `bbb-new-test.sty`, `exam.cls` und `Makefile` (für die Linuxer unter uns) in das lokale Hauptverzeichnis (z.B. `PruefungXY`) kopieren.
5. Erstellung des Hauptdokumentes mit folgender Struktur:

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% PruefungXY.tex
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Dezember 2006 / E. Mösching

\documentclass[10pt,a4paper,german,oneside]{exam}
\usepackage{bbb-new-test}
\usepackage{picins}
\usepackage{float}
\usepackage{multicol}

%%%% Für Lösung auskommentieren
%\printanswers\def\answers{1}

% Ersteller
\author{Ernst Mösching}

% Stichworte
\keywords{}

% Logo
\logo{include/bbbaden}

% Name der Prüfung
\examname{Klausur Drehmomente Serie A}

% Für wen ist diese Prüfung
\class{2. Lehrjahr PMK-E}

% Zeitbedarf
\timespan{1 Lektion}

% Hilfsmittel
\helpstuff{
  \begin{itemize}
    \noVSpace
    \item Taschenrechner
    \end{itemize}
}

% Themen
\topics{
  \begin{itemize}
    \item Drehmomente
    \item Hebel
    \end{itemize}
}

% Prüfungshinweise

```

```

\examnotes{
  \begin{itemize}
  \item Die Lösungen schreiben Sie direkt auf die Aufgabenblätter.
  \item Flüchtige, unsaubere, lückenhafte Darstellung gibt Abzug.
  \end{itemize}
}

% Was sollte der Korrekteur beachten ?
\korrekturhinweise{
  Aufgaben 9: 1 Punkt für die Skizze und 2 Punkte für die Lösung.\\
  Bei Aufgabe 10 gibt es zwei Lösungsmöglichkeiten.
}

\bbbinit

\begin{document}

%% Print the Cover
\maketitle
\clearpage
\pagenumbering{arabic}
\footer{\textcopyright{ } BerufsbildungBaden, \number\year}{Seite
\thepage\ von \numpages}{\today}

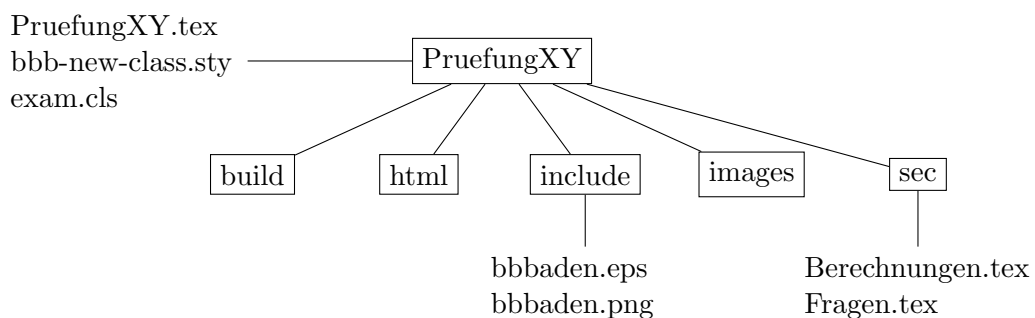
\begin{questions}
  \input{sec/Fragen.tex}
  \clearpage
  \input{sec/Berechnungen.tex}
\end{questions}
\end{document}

```

Alle Fragen befinden sich in der Umgebung `\begin{questions} ... \end{questions}`. Mit `\input{sec/Fragen.tex}` wird z.B. die Datei `Fragen.tex` aus dem Verzeichnis `/sec` eingebunden.

6. Jetzt müssen noch die Dokumente `Fragen.tex` und `Berechnungen.tex` im Verzeichnis `/sec` erstellt werden.

Die Verzeichnisstruktur mit den entsprechenden Dateien sieht dann wie folgt aus:



7. Die Datei `Fragen.tex`

Der Beispiel-Code enthält drei Fragetypen: Multiple Choice, Kurzantworten und Fragen mit ausführlicher Antwort.

Jede Frage beginnt mit `\question[1]`. Die Fragen werden automatisch nummeriert und die Punktezahl wird mit dem Wert in den eckigen Klammern angegeben, beim vorliegenden Beispiel also 1 Punkt.

Die Lösung erfolgt je nach Fragetyp:

Multiple Choice: Mit `\correctchoice`

Kurzantworten: Mit `\answerline` für den Prüfling und für die Lösung in der solution-Umgebung `\begin{solution} ... \end{solution}`

Fragen: `\begin{solution}[2cm] ... \end{solution}` für Leerraum bei den Schülerantworten, bzw. mit `\begin{solutionorlines}[3cm] ... \end{solutionorlines}` für Linien im Leerraum der Schülerantworten. In den eckigen Klammern wird die Größe des Leerraumes angegeben. Die Lösung selbst wird in die solution-Umgebung eingetragen.

Beispiel-Code für Fragen.tex

```

%*****
% Fragen.tex
%*****
% Dezember 2006 / E. Mösching

% Code für Seitenwechsel bei Lösung
% {\ifprintanswers\clearpage\fi}

% Code für Seitenwechsel in der Aufgabenstellung
% {\ifprintanswers\else\clearpage\fi}

% Lösungstext nicht abgesetzt
\renewcommand{\solutiontitle}{\noindent\textbf{Lösung:}\enspace}

% Beispiel für Multiple Choice
\uplevel{\section{Multiple Choice}}

\question[1] Welche Aussage trifft für zwei Kräfte zu, die sich das
  Gleichgewicht halten?
\begin{choices}
  \choice Sie sind gleich gross und schneiden sich.
  \choice Sie sind gleich gross und wirken in gleicher Richtung.
  \correctchoice Sie sind gleich gross, entgegengesetzt gerichtet
    und liegen auf gleicher Wirkungslinie.
  \choice Sie sind gleich gross, stehen rechtwinklig zueinander und
    greifen im gleichen Punkt an.
  \choice Keine Antwort ist richtig.
\end{choices}

% Beispiel für Kurzantworten
\uplevel{\section{Kurzantworten}}

```

```

\question[1] Wie lautet das Hebelgesetz in der allgemeinen
  Formulierung?
\answerline
\begin{solution}
   $\sum M = 0$ 
\end{solution}

```

```

% Beispiel für Fragen mit ausführlicher Antwort
\uplevel{\section{Fragen}}

```

```

\question[2] Geben Sie in Worten und als Formel an, wie das
  Drehmoment berechnet wird.
\begin{solutionorlines}[1.5cm]
  Drehmoment = Kraft mal wirksamer Hebelarm.
   $[F=m \cdot a]$ 
\end{solutionorlines}

```

8. Die Datei `Berechnungen.tex`

Im folgenden Beispiel-Code wird eine Berechnungsaufgabe mit 2 Punkten gestellt. Zur Aufgabenstellung gehört die Zeichnung `Velokurbel`, die sich im Verzeichnis `\images` befindet.

Die Lösung soll direkt im Aufgabenblatt eingetragen werden. Dafür sind 5cm Leerraum vorgesehen.

Beispiel-Code für `Berechnungen.tex`

```

% Berechnungen.tex
% Dezember 2006 / E. Mösching

% Code für Seitenwechsel bei Lösung
% {\ifprintanswers\clearpage\fi}

% Code für Seitenwechsel in der Aufgabenstellung
% {\ifprintanswers\else\clearpage\fi}

\uplevel{\section{Berechnungen}}

\question[2] Berechnen Sie das Drehmoment an der Stelle II für die
  Kraft  $F=200\text{N}$ .\\

\includegraphics{images/Velokurbel}
\begin{solution}[5cm]
   $mm\\
   $M = F \cdot a = 200\text{N} \cdot 127.3\text{mm} = \mathbf{25.46\text{Nm}}$ 
\end{solution}$ 
```

9. Prüfung und Lösung erstellen

Damit alle Referenzen (für die Punktetabelle u.a.) sicher aufgelöst werden können, muss mindestens zweimal mit `pdflatex` kompiliert werden.

Durch das Auskommentieren der Zeile `\printanswers\def\answers{1}` im Beispiel-Code des \LaTeX -Hauptdokumentes wird bestimmt, dass die Lösung erstellt werden soll. Das mit `pdflatex` kompilierte Dokument erhält den Namen des \LaTeX -Hauptdokumentes. Beachten Sie, dass der gleiche Name für die Prüfung wie auch für die Lösung vergeben wird.

Folgendes Vorgehen hat sich gut bewährt:

- Lösung erstellen.
- pdf der Lösung umbenennen, z.B. in `PruefungXY_Loesung.pdf`
- Prüfung erstellen. Der Defaultname, z.B. `PruefungXY.pdf` kann beibehalten werden.
- Wenn nicht mit `Makefile` kompiliert wurde, müssen anschliessend noch beide pdf-Dateien ins Verzeichnis `/build` kopiert werden. Alle temporären Dateien im Hauptverzeichnis können nun gelöscht werden.
Im Hauptverzeichnis sollten sich danach nur noch `*.tex`, `*.sty` und `*.cls` Dateien befinden.

Viele hilfreiche Informationen und Beispiele zur Erstellung einer \LaTeX -Prüfung mit `bbb-new-test.sty` finden Sie im BBB-SVN/Informatik in der Prüfung `modultest_226` unter folgendem Link:
[SVN BBBaden](#)

Literaturverzeichnis

- [1] Jürgens, Manuela: *L^AT_EX - Fortgeschrittene Anwendungen oder: Neues von den Hobbits...*, Fernuniversität Hagen, 1995
- [2] Jürgens, Manuela: *L^AT_EX - eine Einführung und ein bisschen mehr...*, FernUniversität, (ISBN 3-423-30017-5), Hagen, 2000
- [3] Wilkins, David R: *Getting Started with L^AT_EX* , ohne Ort, 1995
- [4] Mittenbach, Franz und Goossens, Michael: *The L^AT_EX Companion - Second Edition*, Addison Wesley, (ISBN 0-201-36299-6), Boston, 2004

Glossar

$\sum M A$: Summe aller Momente um den Lagerpunkt A

F_A : Kraft im Auflager A

Glossar: Eine Wörterliste mit Erklärungen, die meist Bestandteil eines (Fach)textes ist und die Bedeutung von erklärungsbedürftigen Wörtern festlegt.

RFC: Request for Comments. Ein Entwurf mit der Bitte an das Fachpublikum, ihn zu kommentieren.

Index

B	
BBB-Standards	54
BBB-Styles	56
\booland	37
\boolor	37
C	
\cite	18
D	
.DTX Dateien	4
F	
Floats	45
G	
Glossar	44
Grafiken	45
I	
\index	43
Indices	43
Index	43
.INS Dateien	4
L	
\label	17
Listings	38
N	
NewClass	56
\nomenclature	44
P	
Packages	
installation	4
\parpic	46
\piccaption	48
S	
.STY Dateien	4
svn	54
T	
Tabellen	8
W	
\tabitemize	15
thebibliography	18
Wortverzeichnis	<i>siehe</i> Glossar
X	
xr-hyper	18