

README zur Protokollvorlage

Anleitung und Erläuterungen zur LaTeX-Vorlage für Praktikumsprotokolle der physikalischen Anfängerpraktika der Fakultät für Physik, KIT.

v1.1.0 der Anleitung. Bezieht sich auf v1.1.0 der Vorlage.

Inhaltsverzeichnis

1	Einleitung	1
2	Über LaTeX	2
2.1	Kosten und Nutzen von LaTeX	2
2.2	Installation	3
2.2.1	Verwendung einer Virtualbox	3
2.2.2	Installation unter Windows	3
2.2.3	Installation unter Mac OS X	4
2.2.4	Installation unter Linux	4
2.2.4.1	Ubuntu	4
2.2.4.2	ArchLinux und Manjaro	4
2.2.4.3	Fedora	5
2.2.4.4	OpenSUSE	5
2.3	Vor der Installation ist nach der Installation	5
3	Über diese LaTeX-Vorlage	6
3.1	Einige Anmerkungen vorab	6
3.2	Die TeXnische Umsetzung	6
3.2.1	Kompilieren	6
3.2.2	Aufbau	7
3.2.3	Weitere Dateien	8
3.3	Fragen und Verbesserungen	8
4	Nützliche Befehle	9
4.1	Allgemeine Befehlsreferenzen und Dokumentationen	9
4.2	Einige nützliche Beispiele	9
4.2.1	Zitate	9
4.2.2	Anmerkungen	9
4.2.3	Abbildungen und Tabellen	10
4.2.3.1	Abbildungen	10
4.2.3.2	Tabellen	11
4.2.4	Einheiten und wissenschaftliche Notation	12
4.2.5	Weitere Befehle der Vorlage	12
	Literaturverzeichnis	14

1. Einleitung

Das nachfolgende Dokument enthält Erläuterungen und Details zur LaTeX-Vorlage für Praktikumsprotokolle. Es handelt sich dabei um Protokolle für die physikalischen Anfängerpraktika der Fakultät für Physik des KIT. Das Dokument richtet sich an Studierende des KIT, die am Praktikum teilnehmen. Die Vorlage wird von der Fachschaft für Physik bereit gestellt.

Um ein Protokoll gemäß der üblichen Formatierungsrichtlinien erstellen zu können sind zwei Dinge vorausgesetzt. Zum Einen muss der Verfasser sich mit den Richtlinien auskennen. Empfehlenswert sind hierbei der Kurs zum wissenschaftlichen Schreiben vom House of Competence sowie eine kurze Zusammenfassung aller Formatierungsregeln von Herrn Seitz-Moskaliuk, die auf der Seite des Praktikums zu finden sind.

<http://www-ekp.physik.uni-karlsruhe.de/~simonis/praktikum/allgemeines/Protokoll-Formatierungshinweise.pdf>

Die Richtlinien selbst sollen aber nicht Thema dieses Dokuments sein. Zum Anderen muss der Verfasser wissen, wie er diese Richtlinien mit der verwendeten Software umsetzen kann. Im Falle von LaTeX ist das Umsetzen dieser Richtlinien meist enorm einfach, nur der Einstieg in die Verwendung von LaTeX gestaltet sich für viele Studenten nicht so leicht.

Um den Studenten diesen Einstieg so leicht wie möglich zu machen, hat sich die Fachschaft für Physik dazu entschieden, diese Vorlage zur Verfügung zu stellen.

Diese Vorlage setzt die Verwendung des Textsatzsystems LaTeX voraus. Wer sich gegen LaTeX entscheidet, braucht ab dieser Stelle nicht weiter zu lesen. Das Erstellen des Praktikumsprotokolls ohne LaTeX ist durchaus möglich. Mehr Details dazu im nächsten Kapitel.

Zu der Vorlage gibt es ein Beispielprotokoll, welches mithilfe der Vorlage erstellt wurde. Es ist zu beachten, dass diese Vorlage nur ein Beispiel für Textsatz und Formatierungen ist – auf ein formal und inhaltlich korrektes Protokoll wurde nicht geachtet. Musterprotokolle dieser Art können auf den Seiten des Anfängerpraktikums der Fakultät gefunden werden.

2. Über LaTeX

2.1 Kosten und Nutzen von LaTeX

Wer den Kurs *Einführung in das Rechnergestützte Arbeiten* besucht hat, kann vermutlich einen Teil dieses Kapitels überspringen.

LaTeX (Deutsch gesprochen: „Latech“) ist ein Programm, mit dem Textdokumente erstellt werden können, ähnlich wie mit Microsoft Word. Wesentlicher Unterschied ist, dass unter Word direkt das Ergebnis des Geschriebenen sichtbar ist und die Eigenschaften der Darstellung (das sogenannte Markup) durch das Programm verarbeitet und dem Nutzer nicht gezeigt werden. Bei LaTeX muss das Markup vom Nutzer ganz explizit mitverwendet werden, damit das gewünschte Ergebnis entsteht. LaTeX ist unter seinen Nutzern dafür beliebt, dass es äußerst präzise abstimmbare ist und man im Prinzip (im Gegensatz zu z.B. Word) durch nichts außer dem dazugehörigen Aufwand beschränkt ist. Besonders nützlich erweist sich LaTeX beim Setzen von Formeln, weshalb es zum Verfassen naturwissenschaftlicher Dokumente nochmals häufiger verwendet wird. So kann das Setzen folgender Formel in Word etwas mühselig werden:

$$[S, H_1] = \text{h.c.} + \sum_{\substack{kk' \\ \sigma\sigma'}} \frac{V_k V_{k'}}{\epsilon_k - \epsilon_f} \left[c_{k\sigma}^\dagger c_{f\sigma}, c_{k'\sigma'}^\dagger c_{f\sigma'} + c_{f\sigma'}^\dagger c_{k'\sigma'} \right] \quad (2.1)$$

$$+ \sum_{\substack{kk' \\ \sigma\sigma'}} V_k V_{k'} \underbrace{\left(\frac{1}{\epsilon_k - (\epsilon_f + U)} - \frac{1}{\epsilon_k - \epsilon_f} \right)}_{=: 1/A_k} \left[n_{f\bar{\sigma}} c_{k\sigma}^\dagger c_{f\sigma}, c_{k'\sigma'}^\dagger c_{f\sigma'} + c_{f\sigma'}^\dagger c_{k'\sigma'} \right]$$

$$= \text{h.c.} + \sum_{\substack{kk' \\ \sigma\sigma'}} \frac{V_k V_{k'}}{\epsilon_k - \epsilon_f} (c_{k\sigma}^\dagger c_{k'\sigma} - \delta_{kk'} n_{f\sigma}) \quad (2.2)$$

$$- \sum_{\substack{kk' \\ \sigma\sigma'}} \frac{V_k V_{k'}}{A_k} (c_{k\sigma}^\dagger c_{f\sigma} (c_{k'\bar{\sigma}}^\dagger c_{f\bar{\sigma}} + c_{k'\bar{\sigma}}^\dagger c_{f\bar{\sigma}}) + n_{f\bar{\sigma}} (\delta_{kk'} n_{f\sigma} - c_{k\sigma}^\dagger c_{k'\sigma}))$$

$$= \underbrace{-2 \sum_{k\sigma} \left(\frac{V_k^2}{\epsilon_k - \epsilon_f} n_{f\sigma} + \frac{V_k^2}{A_k} n_{f\bar{\sigma}} \right)}_{=: H_0^\Delta} + \underbrace{\sum_{\substack{kk' \\ \sigma\sigma'}} \frac{V_k V_{k'}}{A_k} (c_{k\sigma}^\dagger c_{k'\bar{\sigma}}^\dagger c_{f\sigma} c_{f\bar{\sigma}} + \text{h.c.})}_{=: H_{\text{pair}}} \quad (2.3)$$

$$+ \underbrace{\sum_{\substack{kk' \\ \sigma\sigma'}} \left(\frac{V_k V_{k'}}{\epsilon_k - \epsilon_f} c_{k\sigma}^\dagger c_{k'\sigma} - \frac{V_k V_{k'}}{A_k} (c_{f\bar{\sigma}}^\dagger c_{f\sigma} c_{k\sigma}^\dagger c_{k'\bar{\sigma}} - n_{f\bar{\sigma}} c_{k\sigma}^\dagger c_{k'\sigma}) + \text{h.c.} \right)}_{=: H_{\text{int}}}.$$

Einzelne Feinheiten bei der Formelsetzung sind unter Word dazu gar nicht möglich, LaTeX ist hier hingegen sehr flexibel und es gibt praktisch keine Einschränkungen. LaTeX ist zu dem OpenSource und für jedermann frei zugänglich. LaTeX wird außerdem in einigen

Onlineportalen als Möglichkeit zum Schreiben mathematischer Formeln unterstützt. LaTeX verfügt weiter über viele Funktionen, die zum Erstellen und Verwalten von Inhalts-, Abbildungs-, Tabellen- sowie Literaturverzeichnissen nützlich sind. Die Erweiterung BibTeX stellt hier eine wesentliche Hilfe beim Verwalten von Zitaten und einer umfangreichen Literaturlatenbank dar.

Die Fachschaft und viele Dozenten empfehlen wegen dieser und auch wegen weiterer Gründe den Einstieg in LaTeX, er ist jedoch nicht zwingend notwendig. Wer seine Protokolle mit Word und dergleichen verfassen möchte, dem steht dies absolut zu.

2.2 Installation

2.2.1 Verwendung einer Virtualbox

In den nachfolgenden Abschnitten wird die Installation von LaTeX unter verschiedenen Betriebssystemen behandelt. Alternativ zur lokalen Installation von LaTeX auf dem eigenen Rechner kann man LaTeX auch in einer virtuellen Maschine nutzen. Eine virtuelle Maschine ist quasi ein vollständiges Betriebssystem, das als Programm auf dem lokalen Rechner läuft und bei Bedarf aufgerufen werden kann. Prof. Quast vom Institut für exp. Kernphysik stellt eine solche virtuelle Maschine zur Verfügung. Sie basiert auf Kubuntu 14.04 und stellt neben vielen anderen Programmen auch die *TeX Live* Pakete aus den Ubuntu-Quellen zur Verfügung. Der Vorteil einer virtuellen Maschine ist, dass trotz der vielen verschiedenen Betriebssysteme und Programme, die im Privatbereich zum Einsatz kommen, eine vorgefertigte und immer gleich bleibende Arbeitsumgebung für Studenten geschaffen werden kann, die verlässlich läuft und den Studenten alle notwendige Software zum Arbeiten zur Verfügung stellt. Besonders Studenten, die die Installation einer Linux-Distribution auf ihrem eigenen Rechner gänzlich scheuen (sowohl die vollständige Ersetzung von Windows als auch das Aufsetzen eines Dualboot-Systems), können dadurch leicht Linux ausprobieren und sich damit vertraut machen, ohne von dem Betriebssystem grundsätzlich abhängig zu sein und es vollständig installieren zu müssen.

Anleitungen zu seiner Virtualbox stellt Prof. Quast vom IEKP auf seiner Seite zur Verfügung (<http://www-ekp.physik.uni-karlsruhe.de/~quast/VMroot/>). Man lese dazu die Anleitungen „VMachine.pdf“ und „VMroot.pdf“. In der Virtualbox ist *TeX Live* von 2012 aus den Ubuntuquellen installiert. Damit kann die Vorlage komplett kompiliert werden. Als LaTeX-Editor empfiehlt sich Kile, welcher ebenfalls bereits vorinstalliert ist.

2.2.2 Installation unter Windows

Es gibt im Wesentlichen zwei Editoren, die sich für das Bearbeiten von TeX-Dateien unter Windows eignen, nämlich *MikTeX* (<http://www.miktex.org/>) und *Texmaker* (<http://www.xmlmath.net/texmaker/>). Beide bringen LaTeX von Hause aus mit und zusätzlich benötigte Pakete werden automatisch nachinstalliert. Ergänzend zu *MikTeX* kann man auch den TeX-Editor TeXnicCenter (<http://www.texniccenter.org/>) nutzen, der extra für das Zusammenspiel mit *MikTeX* ausgelegt ist.

Alternativ kann man *TeX Live* installieren, die zurzeit umfangreichste LaTeX-Distribution. Infos zur Installation der aktuellen *TeX Live* Version findet man ebenfalls auf der offiziellen *TeX Live* Website: <https://www.tug.org/texlive/>. Einen Überblick über weitere Editoren speziell auch für Windows und Mac bietet Herr Dr. Poenicke auf seiner Seite zum ERA-Kurs: http://comp.physik.kit.edu/Lehre/ERA/05_Latex/.

Zum Kompilieren der Vorlage unter *MikTeX* ist es evtl. nötig, diverse Pakete manuell nachzuinstallieren. Dazu öffnet man den PackageManager (im Startmenü unter MiKTeX → Maintenance (admin) → PackageManager(admin) wählen). Dort muss im PullDownMenue „Repository“ der Befehl „synchronize“ ausgeführt werden.

2.2.3 Installation unter Mac OS X

Unter Mac OS X steht, ähnlich wie mit *TeX Live* unter Linux, mit *MacTeX* eine sehr umfangreiche Distribution zur Verfügung und sogar auch eine abgespeckte Version, falls die SSD doch zu klein ist. Das 2.4 GB große *pkg* für OS X ab 10.5 und sogar für noch ältere Versionen gibt es unter <https://www.tug.org/mactex/> immer aktuell. Ein *pkg* zu installieren ist zwar aufwendiger als ein *DMG* zu installieren, aber mit ein paar Klicks auch schnell geschehen. Natürlich gibt es für die Freunde des Terminals auch ein Homebrew-Cask mit Namen *mactex*.

Eine weiter auf *TeX Live* basierende Lösung für OS X stellt *TeXShop* (<http://pages.uoregon.edu/koch/texshop/>) dar. Dieser Editor bietet unter anderem eine Live-Vorschau, ist jedoch von der Benutzeroberfläche nicht jedermanns Sache. Als Alternative bietet sich hier natürlich auch wieder *Texmaker* an (<http://www.xmlmath.net/texmaker/>).

2.2.4 Installation unter Linux

TeX Live stellt eine umfangreiche LaTeX-Distribution dar, die entweder direkt von der offiziellen *TeX Live*-Website oder (was meist einfacher und in vielen Fällen ausreichend ist) über spezielle dafür vorgesehene Pakete der einzelnen Linux-Distribution installiert werden kann (s.u.). Unabhängig von der Installation einer LaTeX-Distribution benötigt man einen entsprechenden TeX-Editor. Hier bieten sich an: Kile (KDE), Texmaker (QT), Eclipse (Java, über das Plugin TeXlipse) oder JLatexEditor (Java). Siehe auch: <http://wiki.ubuntuusers.de/LaTeX-Editoren>.

2.2.4.1 Ubuntu

Möchte man *TeX Live* aus den Ubuntu-Quellen installieren, so sind folgende Pakete meist ausreichend (z.B. für das Bearbeiten der Protokollvorlage):

```
texlive texlive-lang-german texlive-doc-de
texlive-latex-extra texlive-science
```

Möchte man sich vor lästigem Neuinstallieren weiterer Pakete schützen und dafür eine etwas zeit- und speicherplatzintensivere Installation in Kauf nehmen, bietet sich die Installation des Pakets `texlive-full` an. Die derzeitige *TeX Live* Version in den Ubuntu-Quellen ist von 2012. Die aktuellste *TeX Live* Version ist von 2014. In den meisten Fällen kommt man mit der Version von 2012 gut aus. Informationen zur manuellen Installation findet man auf der offiziellen *TeX Live*-Website: <https://www.tug.org/texlive/>. Entscheidet man sich dazu, die aktuellsten *TeX Live* packages manuell zu installieren, handelt man sich häufig Probleme mit der Paketverwaltung ein, wenn man versucht TeX-Editoren, wie z.B. Kile zu installieren. Eine Lösung findet man für Kile (und analog auch für andere Editoren) z.B. unter <http://tex.stackexchange.com/questions/63302/kile-and-texlive-2012>, hier für das gleiche Problem bei der Umstellung von Version 2009 auf 2012.

2.2.4.2 ArchLinux und Manjaro

Die Archlinux-Quellen enthalten stets die aktuellste *TeX Live*-Version. Eine vollständige *TeX Live* Installation erreicht man mit:

```
pacman -S texlive-most
```

2.2.4.3 Fedora

Wie unter Ubuntu so kommt mit Fedora *TeX Live* nicht in der aktuellsten Version mit. Allerdings ist die Version 2013 in Fedora 20 die vorletzte Version. Sie ist in viele einzelne Pakete aufgeteilt, die alle mit `texlive` beginnen und bei Bedarf nachinstalliert werden können. Selbstverständlich klappt die Installation direkt aus den Quellen wie unter Ubuntu gut, jedoch muss man beim Nachinstallieren von Schriften gegebenenfalls als Benutzer `root` arbeiten und ein `sudo ...` reicht nicht aus. Mit dem folgenden Befehl installiert man die wichtigsten Pakete unter Fedora:

```
yum install texlive-scheme-basic texlive-collection-mathextra  
texlive-collection-science texlive-collection-langgerman
```

2.2.4.4 OpenSUSE

Ein vollständige *TeX Live* Installation unter OpenSUSE funktioniert mit:

```
zypper install texlive
```

2.3 Vor der Installation ist nach der Installation

Wer nun LaTeX installiert hat, hat zwar viele mächtige Textsatz-Programme auf seinem Rechner laufen, mit denen er jedoch zunächst nichts anfangen kann, wenn er nicht bereits Erfahrung mit LaTeX hat. Bevor man mit dem Kompilieren und Bearbeiten der Protokollvorlage beginnt, ist es sicher hilfreich, sich zunächst ein paar Tutorials und Anleitungen sowie Minimalbeispiele zu LaTeX anzuschauen und auszuprobieren. Diese findet man zahlreich im Internet und die zu Beginn von Kapitel 4 angegebenen Seiten könnten ebenfalls nützlich sein.

Es ist auch nicht unbedingt notwendig, sein eigenes Protokoll nur unter der Verwendung der Protokollvorlage zu schreiben. Wichtig ist der Lernerfolg beim Schreiben des Protokolls und nicht das sture Verwenden der Vorlage. Wem die Vorlage zu Beginn zu mächtig ist und wer Probleme damit hat, sie zu verstehen, kann seine eigene Vorlage „from scratch“ schreiben und verwenden und ggf. später auf die Vorlage umsteigen. Umgekehrt darf jeder die Vorlage modifizieren und nach den eigenen Wünschen anpassen und vielleicht, nachdem etwas Erfahrung mit LaTeX gesammelt wurde, eine eigene Vorlage entwerfen und benutzen.

3. Über diese LaTeX-Vorlage

3.1 Einige Anmerkungen vorab

Dieses Dokument über die LaTeX-Vorlage wurde seinerseits mit der Protokollvorlage erstellt (nur das Deckblatt wurde weggelassen).

Prinzipiell ist die Vorlage für das beidseitige Drucken gedacht.

Überlicherweise unterteilt man wissenschaftliche Berichte in Kapitel, bei denen jedes Kapitel auf einer ungeraden Seite beginnt. Bei einem Praktikumsbericht von ein paar wenigen Seiten handelt es sich hingegen schlicht um Papierverschwendung. Standardmäßig werden daher keine leeren Seiten eingefügt. Der Nutzer der LaTeX-Vorlage hat hier die freie Wahl, ob er danach verfahren möchte oder nicht. Falls nicht, kann über den Befehl `\cleardoublepage` am Ende eines Kapitels das Neueröffnen der folgenden Kapitels auf einer ungeraden Seite erzwungen werden. Mit dem Befehl `\blankpage` kann sogar eine komplette leere Papierseite eingefügt werden.

Standardmäßig ist die Seitennummerierung fortlaufend. Es gibt zwei zusätzliche Optionen, die eine kapitelweise Nummerierung, bzw. einen Neubeginn der Seitenzahl mit jedem Kapite ermöglichen. Mehr dazu weiter unten.

3.2 Die TeXnische Umsetzung

3.2.1 Kompilieren

Die Datei `main.tex` muss mit PDFLaTeX kompiliert werden, um die druckbare PDF-Datei `main.pdf` zu erstellen. Ein vollständiger Kompilier-Befehl umfasst dabei 3 Mal PDFLaTeX und 1 Mal BibTeX (falls BibTeX verwendet wird), und zwar in der folgenden Reihenfolge:

```
pdflatex main.tex
bibtex main.aux
pdflatex main.tex
pdflatex main.tex
```

Falls die LaTeX-Zwischendateien (`*.aux`, `*.toc`, `*.out`, etc.) vorhanden sind, reicht für kleine Änderungen in der Regel ein Durchlauf von PDFLaTeX. Viele Editoren entscheiden selbst, wie häufig womit kompiliert werden muss. Es reicht dann eine Kompilierung mit PDFLaTeX zu befehlen.

Eine Alternative zu der beschriebenen Befehlsfolge ist das Programm `latexmk`, das unabhängig vom Texteditor im Terminal verwendet werden kann. Es ist entweder mit der TeX-Distribution mitgeliefert worden, oder kann über die entsprechende Paketverwaltung von TeX oder dem Betriebssystem installiert werden. Im Terminal mit `latexmk -pdf main.tex`

aufgerufen, ermittelt es selbstständig, welche Befehle wie oft und in welcher Reihenfolge ausgeführt werden müssen. Mit dem zusätzlichen Parameter `-pvc` läuft `latexmk` ständig weiter und kompiliert immer neu, wenn sich eine der ins Dokument eingebundenen Dateien ändert: `latexmk -pdf -pvc main.tex`.

3.2.2 Aufbau

In der Datei `main.tex` werden alle Einstellungen am Dokument und die eingebundenen Kapitel definiert. Die Datei basiert auf einer speziell für das Praktikum angelegten Dokumentenklasse, die sich im Ordner `include/` mit dem Namen `protokollclass.cls` befindet. Diese definiert das Layout des Protokolls.

Der erste größere Block am Anfang der Datei besteht nur aus der Dokumentenklasse (`\documentclass`) und einer Aufzählung der zum Dokument gehörigen Dateien sowie einiger nützlicher Pakete, die die Protokollklasse zur Verfügung stellt.

Beim zweiten Block können die Daten zum vorliegenden Protokoll eingetragen werden. Die Funktion der Befehle können den Kommentaren entnommen werden.

Der dritte Block mit der Überschrift **Settings for word separation** dient dazu, LaTeX zu vermitteln, wie Wörter zu trennen sind, die LaTeX von Hause aus nicht kennt. Meist kommt LaTeX auch ohne dies klar, aber hin und wieder ergeben sich unschöne Probleme bei sehr langen Wörtern, die man lieber vermeiden möchte.

Der vierte Block beinhaltet einige auskommentierte Befehle, die bei Bedarf eingeschaltet werden können. Diese Befehle sind in der Datei kommentiert. Die letzten beiden Befehle beeinflussen die Seitennummerierung des Dokuments. Standardmäßig ist die Seitennummerierung fortlaufend.

Falls ein Protokoll auf Englisch verfasst werden soll, können vordefinierte Begriffe wie Literaturverzeichnis (Engl.: Bibliography) oder Kapitel (Engl.: Chapter) auf eine gewählte Sprache eingestellt werden. Diese Sprache wird auch im Literaturverzeichnis verwendet. Diese Einstellung erfolgt mit dem Befehl `\SelectLanguage{lang}`.

Nach dem Befehl `\begin{document}` wird der eigentliche Inhalt des Dokuments festgelegt. Zuerst werden die Titelseite eingebunden und ein Inhaltsverzeichnis erstellt. Der Befehl `\FrontMatter` sagt, dass der nachfolgende Teil im PDF-Verzeichnis mit römischen Zahlen nummeriert werden soll und stellt weitere Layoutdetails ein.

Überlicherweise beginnt das Literaturverzeichnis eines Berichtes auf einer ungeraden Seite und die Rückseite des Deckblatts ist leer. Da nun Praktikumsprotokolle vielleicht nicht den höchsten Anspruch erheben müssen und bei den vielen Protokollen pro Jahr einiges an Papier gespart werden kann, ist es vielleicht empfehlenswert sich für die kürzere Version mit Literaturverzeichnis auf der Rückseite des Deckblatts zu entscheiden, auch wenn es normalerweise stilistisch anders gehalten wird. Eine leere Rückseite kann jederzeit mit `\cleardoublepage` nach dem Einbinden der Titelseite eingefügt werden.

Mit dem Befehl `\MainMatter` werden Seitennummerierung, Layout, etc. auf den Beginn des eigentlichen Inhalts des Dokuments eingestellt. Es folgen die verschiedenen Kapitel, die mit `\chapter{Kapitelname}` definiert werden. Der Inhalt wird aus `.tex`-Dateien mit Hilfe von `\input{...}` eingelesen. Auch hier können Kapitel wieder auf geraden sowie auf ungeraden Seiten eröffnen. Wer damit nicht zufrieden ist, kann mit `\cleardoublepage` nach dem `\input`-Befehl nachhelfen.

Der Anhang dürfte für die meisten Protokolle nicht notwendig sein, man kann ihn jedoch sehr leicht erzeugen. `\Appendix` stellt wieder das Layout um und die nachfolgenden

Befehle erzeugen ein unnummeriertes Kapitel *Anhang*, welches wieder über `\input{...}` eingebunden wird.

Mit dem Literaturverzeichnis tun sich viele Studenten im Praktikum schwer, da das meiste ohnehin nur aus der Vorbereitungsmappe stammt. Dennoch stellt die Vorlage sowohl ein einfaches `\TheBibliography`- sowie ein umfangreiches Bibtex-Literaturverzeichnis zur Verfügung. Der Befehl `\TheBibliography` dient wieder zum Verstellen des Layouts. (Achtung: die Namensgebung ist irreführend! Dieser Befehl hat nichts der eigentlichen `thebibliography`-Umgebung zu tun.) Danach kann man mithilfe von Bibtex über den Befehl `\bibliography{file.bib}` ein Literaturverzeichnis erzeugen. Die Datenbank muss dabei im Bibtex-Format in der Datei `file.bib` vorliegen. Ein Beispiel befindet sich in der Datei `lit.bib`. Das Layout des Literaturverzeichnisses legt man über den Befehl `\bibliographystyle{style}` fest. In dieser Vorlage wird der Stil `babalpha` verwendet. Es können auch alle weiteren kompatiblen Stile verwendet werden. Eine Einführung in die verschiedenen Zitierstile erfolgt hier nicht. Weitere kompatible Babelbib-Style kann man online suchen. Alternativ zum Bibtex-Literaturverzeichnis kann man ein simpleres Literaturverzeichnis mit dem Befehl `\TheBibliography` erstellen, welcher von Hause aus mit LaTeX kommt. Die Syntax ist im Dokument erläutert.

3.2.3 Weitere Dateien

Einige Kurzbefehle wurden in der Datei `include/cmds.tex` angelegt, weitere können hier nach Belieben hinzugefügt werden. Die Datei `titelpage.tex` selbst sollte aus organisatorischen Gründen nach Möglichkeit nicht bearbeitet werden, damit das Layout der Titelseite für alle abgegebenen Protokolle der verschiedenen Gruppen gleich bleibt. Die Inhalte können, wie erwähnt, über die Einträge am Anfang der Datei `main.tex` angepasst werden.

Bilder und Tabellen können natürlich von überall aus eingebunden werden, aber es empfiehlt sich aus Gründen der Übersicht, für diese Dateien ebenfalls Unterordner anzulegen. Häufig verwendet man die Bezeichnungen `fig` für den Unterordner mit den Abbildungen und `tab` für den Unterordner mit den Tabellen.

Die Dateien `LICENSE` und `README.md` wird für die Vorlage nicht benötigt. Die Makefile kann von `make` verwendet werden, um die PDFs zu kompilieren. Wer einen LaTeX-Editor benutzt, braucht diese Datei ebenfalls nicht.

3.3 Fragen und Verbesserungen

Diese LaTeX-Vorlage soll als Anregung und vereinfachten Einstieg in das Schreiben von Praktikumsprotokollen mit LaTeX dienen. Für das Praktikum darf (und am besten soll) jeder Praktikant die Vorlage verändern und Dinge ausprobieren, um den Umgang mit LaTeX zu lernen. Einzig das Deckblatt soll nicht umgestaltet werden, damit es keine Probleme bei der Verwaltung des Praktikums gibt. Verbesserungsvorschläge (hier auch gerne zur Titelseite), aber natürlich auch Fragen können jederzeit an die Fachschaft an die Mailadresse `latexvorlage@fachschaft.physik.kit.edu` gerichtet werden.

Es ist bekannt, dass die gleichzeitige Verwendung des `microtype` und des `siunitx` Paket Warnungen ausgibt. Das Problem besteht darin, dass `siunitx` Befehle neu definiert, die `microtype` selbst mitbringt. Dabei entsteht kein deutlich sichtbares Problem und die PDF kann nachwievor problemlos kompiliert werden. Das Problem wird vermutlich in eine der nächsten Versionen der Vorlage gelöst.

4. Nützliche Befehle

4.1 Allgemeine Befehlsreferenzen und Dokumentationen

Die LaTeX-Kurzeinführung von W. Schmidt und Anderen ist kompakt und sehr hilfreich: <ftp://ftp.fu-berlin.de/tex/CTAN/info/lshort/german/l2kurz.pdf>.

Benötigt man allgemeine Hilfe zu einem bestimmten Thema, wird man oft hier fündig:

- <https://en.wikibooks.org/wiki/LaTeX>
- <https://de.wikibooks.org/wiki/LaTeX-Kompendium>,
- <https://de.wikipedia.org/wiki/Hilfe:TeX>,
- <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>.

Möchte man Hilfe zu einem bestimmten vorliegenden Problem erhalten, so kann man unter <https://tex.stackexchange.com/> ganz gezielt suchen oder auch Fragen stellen.

Sucht man nach dem Befehl für ein ganz bestimmtes Zeichen, ist folgendes Dokument nützlich: www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf. Oder folgende Website: <http://detexify.kirelabs.org/classify.html>

4.2 Einige nützliche Beispiele

4.2.1 Zitate

Zitate erstellt man mit dem Befehl `\cite{key}`, wobei `key` die Kurzbezeichnung im Literaturverzeichnisses ist. Das Ergebnis ist dann z.B.: [EKS07]. Möchte man auf ein spezielles Detail einer Quelle verweisen, so lässt sich dies über `\cite[optionales]{key}` bewerkstelligen. Dies erzeugt dann: [EKS07, S. 42].

4.2.2 Anmerkungen

Möchte man sich eine Erinnerung anlegen, was später noch erledigt werden muss, kann der von der Vorlage definierte Befehl `\todo{Anmerkung}` nützlich sein. Als Beispiel:

Der Wert für die gemessene Magnetisierung M des Eisenkerns nach wiederholter Magnetisierung und Entmagnetisierung beträgt [Wert] bei 25 °C. Diesen Werte und weitere Vergleichswerte anderer Quellen befinden sich auch in Tab. [Verweis einf.]

ToDo
ToDo

Hilfreich kann auch die Umgebung `\begin{deprecated}...\end{deprecated}` sein, mit der man ganze Absätze, die man im Dokument nicht mehr haben will, aber zunächst noch (vielleicht als Platzhalter) aufbewahren möchte, geeignet kennzeichnen kann. Als Beispiel:

Dieser Text ist noch da, aber er ist eigentlich nicht sehr sinnvoll und sollte besser ersetzt oder gründlich überarbeitet werden.

4.2.3 Abbildungen und Tabellen

Dieses Thema ist besonders umfangreich, aber auch sehr wichtig für jeden Praktikanten, daher dennoch ein paar ganz explizite Hilfestellungen beim Einbinden von oben genanntem. Diese Anleitungen sind bei weitem nicht vollständig und detailliert. Mehr Infos findet man jedoch auf den oben weiter aufgeführten Seiten.

4.2.3.1 Abbildungen

Als Beispiel:

```
\begin{figure}[tb]
  \centering
  \includegraphics[scale=0.8]{fig/vorbereitung_abbildung123.eps}
  \xcaption{Absorption von Alpha-Teilchen:}{Detaillierte Beschreibung\
    von der zu sehenden Abbildung (aus \cite[S. 507]{dem}).}
  \label{fig:vorbereitung_abbildung123}
\end{figure}
```

Die `figure`-Umgebung ist zum Einbinden von Abbildungen gedacht. Der Befehl `\centering` zentriert die Abbildung mittig auf der Seite. Mit `\includegraphics{...}` wird die eigentliche Abbildung als Datei eingebunden, dabei ist die Angabe des Pfades aus Sicht von `main.tex` zu wählen. Der Teil `[scale=0.8]` ist optional und kann weggelassen werden. Die Zahl hinter dem Gleichheitszeichen ist eine Fließkommazahl und kann modifiziert werden, um die Größe des Bildes zu ändern. Mit dem Befehl `\xcaption{<short description>}{<long description>}` wird eine Bildunterschrift erzeugt. Es ist möglich, den Befehl oberhalb von `\includegraphics{...}` einzufügen, um so ein Bildüberschrift zu erzeugen. Auch wenn es nicht Ziel dieses Textes ist, über die Formalien beim Schreiben eines wissenschaftlichen Textes aufzuklären, sei dennoch schnell angemerkt: Man hat stets **Abbildungen**unterschrift und **Tabellen**überschriften zu verwenden.

Der Befehl `\label{key}` ist ebenfalls optional und dient zum Verweisen auf eine Abbildung. So kann mit Hilfe von `\ref{key}` die Nummer der Abbildung erzeugt werden. Somit muss man bei einem Verweis nicht immer etwas ändern, falls sich die Nummer der Abbildung ändert. Als **key** kann grundsätzlich alles verwendet werden, nur mit einigen Sonderzeichen muss man vorsichtig sein. Der **key** sollte außerdem eindeutig sein, d.h. er sollte nicht für mehrere Abbildungen zugleich verwendet werden. Es ist üblich den Schlüssel stets mit **fig:**, **tab:** oder **eq:** zu beginnen, je nach dem worum es sich in dem Verweis gerade handelt. Dies ist jedoch nicht zwingend notwendig.

Die Buchstaben **tb** in eckigen Klammern nach `\begin{figure}` dient zur Angabe der Positionierung auf der Seite. LaTeX übernimmt das Anordnen der Abbildungen und Tabellen selbst. In diesem Fall würde LaTeX versuchen die Abbildung zunächst an den Anfang der Seite (**top**) und, falls das nicht geht, an das Ende der Seite (**bottom**) zu setzen. Viele Studenten neigen dazu, LaTeX mit Hilfe der Option **h** oder noch stärkeren Befehlen dazu zwingen zu wollen, Abbildungen auf die Mitte der Seite zu setzen (an den Ort, von wo aus gerade verwiesen wird). Dies ist ebenfalls unter wissenschaftlichen Texten unüblich. Abbildungen sollten stets am Anfang oder am Ende einer Seite stehen. Möchte man sicher stellen, dass gewisse Abbildungen bis zu einem bestimmten Punkt eingebunden sind und nicht in einer viel späteren **section** erscheinen, kann man an dieser Stellen den Befehl `\FloatBarrier` verwenden.

Als Bildformat kommen die meisten gängigen Dateiformate in Frage, wie z.B. **pdf**, **ps**, **eps**, **png**, **bmp** oder **jpg**. Es empfiehlt sich natürlich stets die Verwendung von Vektorgrafiken

der von Rastergrafiken vorzuziehen, sofern dies möglich ist. Man kann eine Abbildung auch ohne Endung angeben und mit verschiedenen Endungen im entsprechenden Ordner hinterlegen und LaTeX sucht sich aus, welches Format es nutzen möchte.

Der Befehl `xfigure`, den die Vorlage in der Datei `./include/cmds.tex` mit sich bringt, fasst die obigen Befehle in einem kurzen Befehl mit vielen Optionen zusammen:

```
\xfigure{<pos>}{<scale>}{<name of file>}{<short caption>}{<long caption>}
```

Der name der Datei (im Ordner `./fig/` und zwar ohne seine Endung; LaTeX bestimmt die Endung selbst) dient auch als Referenzname in der Form `\ref{fig:<name of file>}`. Damit lassen sich die obigen Befehle kurz schreiben:

```
\xfigure{tb}{0.8}{vorbereitung_abbildung123}\
    {Absorption von Alpha-Teilchen:}\
    {Detaillierte Beschreibung von der zusehenden Abbildung ...}
```

4.2.3.2 Tabellen

Als Beispiel:

```
\begin{table}[tb]
  \centering
  \caption{Beschreibung der Tabelle.}
  \label{tab:vorbereitung_tabelle123}
  \rowcolors{3}{gray!10}{white}
  \begin{tabular}{rrr}
    \toprule
    Spalte 1 & Spalte 2 & Spalte 3 \\
    \midrule
    1 & 2 & 3 \\
    4 & 5 & 6 \\
    7 & 8 & 9
    \bottomrule
  \end{tabular}
\end{table}
```

Die Befehle sind fast alle dieselben wie bei der `figure`-Umgebung, hier taucht jedoch statt dem `\includegraphics{...}`-Befehl die `tabular`-Umgebung auf. Es ist manchmal hilfreich und nützlich, diese Umgebung in eine separate Datei im Ordner `tab` zu verlagern und im Dokument die Umgebung durch `\input{tab/datei.tex}` einzubinden.

Der Parameter in geschweiften Klammern hinter dem Beginn der Umgebung gibt die Zahl und die Orientierung der Spalten an (hier sind es drei und sie sind alle rechtsbündig orientiert). Von einer Spalte zur nächsten springt man mit dem Kaufmannsund (`&`) und mit zwei Backslashes (`\\`) springt man in eine neue Reihe.

Der Befehl `\rowcolor...` dient zur Einfärbung verschiedener Zeilen, was die Übersichtlichkeit fördert. Durch einfügen eines senkrechten Striches (aka. Pipe: `|`) zwischen den Angaben zur Orientierung der Spalten können senkrechte Linien in den Tabellen erzeugt werden. Durch `\hline` können waagerechte Striche zwischen verschiedenen Reihen eingefügt werden. Auch hier wird wieder darauf verwiesen, dass üblicherweise eine Tabelle nur über

die in obigem Minimalbeispiel auftauchenden waagerechten Trennlinien über und unter der ersten Zeile sowie am Ende verwendet werden. Alles andere verwirrt eher und sorgt nicht für mehr Übersicht. Die Linien werden hier mit den erweiterten Befehlen `\toprule`, `\midrule` und `\bottomrule` aus dem `booktabs`-Paket verwendet. Diese stellen zusätzlich die Linienbreite und die Abstände ein.

Auch hier gibt es wieder einen verkürzten Befehl:

```
\xtable{<pos>}{<name of file>}{<short caption>}{<long caption>}
```

Hier ist wieder der Name der `.tex`-Datei im Ordner `./tab/` ohne die Dateierweiterung selbst zu verwenden.

4.2.4 Einheiten und wissenschaftliche Notation

Einheiten unterliegen typographischen Regeln, deren Einhaltung das Paket `siunitx` komfortabel möglich macht. Außerdem beinhaltet es ein paar interessante Zusatzfunktionen. Einheitensymbole werden mit aufrechten Kleinbuchstaben gesetzt, Ausnahmen bestätigen die Regel: ist die Einheit von einem Namen abgeleitet, wird der erste Buchstabe groß geschrieben. Außerdem darf man für Liter auch ein großes L verwenden, damit es nicht mit einer Eins verwechselt wird.

`\SI{5,4}{\metre}` setzt 5,4 m mit richtigem Abstand zwischen Zahl und Einheit. Außerdem verhindert es die Trennung zwischen den beiden. `\si{}` setzt nur eine Einheit, `\num{}` nur eine Zahl. Einheiten werden mit Befehlen gesetzt, zum Beispiel `\second`, `\metre`, `\degree`. Präfixe stehen ebenfalls zur Verfügung: `\kilo`, `\centi`, `\micro`. Für Potenzen gibt es `\square`, `\squared`, `\cubic`, `\cubed`, `\raiseto{}` und `\tothe{}`. Negative Potenzen erreicht man mit `\per`. Außerdem ist es möglich, Fehler gleich mit anzugeben: `\SI{5,4(12)}{\metre}` ergibt $(5,4 \pm 1,2)$ m. Wissenschaftliche Notation ist auch möglich: `\SI{5,4(12)e-3}{\kilo\metre}` ergibt $(5,4 \pm 1,2) \cdot 10^{-3}$ km.

Was als Dezimaltrennzeichen benutzt wird, und wie die Fehler gesetzt werden, definieren Optionen, die entweder bei jedem Objekt mitgegeben werden oder bereits in der Präambel gesetzt werden. Speziell diese Vorlage übernimmt diese Arbeit und zwar abhängig von der Option `\SelectLanguage`, die in der `main.tex`-Datei definiert wird.

Eine grobe Übersicht erhält man durch: http://www.suedraum.de/latex/stammtisch/werte_und_einheiten_siunitx.pdf. Die detaillierte Paketbeschreibung gibt es hier: <http://mirrors.ctan.org/macros/latex/contrib/siunitx/siunitx.pdf>.

4.2.5 Weitere Befehle der Vorlage

In der Datei `include/cmds.tex` finden sich einige zusätzliche Befehle, die die Vorlage mit sich bringt und die sich oft als sehr nützlich erweisen. Ein paar Befehle werden hier aufgelistet:

Befehl	Verwendung
<code>\d</code>	Differenzielles d, z.B. $\int dx$
<code>\arr{alg}{...}</code>	Inhalt in ... wird entsprechend der Spalten im ersten Argument ausgerichtet (z.B. <code>ccc</code>). Zu verwenden in eine Mathematik-Umbgebung.
<code>\op{ArCoTanh}</code>	Darstellung von Funktionen oder Operatoren (mit aufrechter Schrift). Zu verwenden bei Funktionen, die nicht bereits definiert sind (also nicht bei <code>\exp</code> oder <code>\sin</code>).
<code>\pdb{f}{x}</code>	Ergibt $\frac{\partial f}{\partial x}$.
<code>\fullstop, \comma</code>	Zeichensetzung in Mathematikumgebung.
<code>\cre, \anh, \numb</code>	Erzeugungs-, Vernichtungs-, Zähloperator in der Quantenmechanik (z.B. c_k^\dagger, c_k, n_k).
<code>\bra{\Psi}</code>	$\langle \Psi $
<code>\ket{\Psi}</code>	$ \Psi\rangle$
<code>\braket{\Psi}{\Phi}</code>	$\langle \Psi \Phi \rangle$
<code>\braenket{\Psi}{T}{\Phi}</code>	$\langle \Psi T \Phi \rangle$
<code>\anglemean{A}</code>	$\langle A \rangle$
<code>\norm{\Psi}</code>	$\ \Psi\ $
<code>\updownarrows</code>	\updown
<code>\downuparrows</code>	\downup
<code>\neswarrows</code>	\nearrow
<code>\swnearrows</code>	\swarrow

Literaturverzeichnis

- [EKS07] Eichler, H. J., H. D. Kronfeldt und J. Sahm: *Das Neue Physikalische Grundpraktikum*. Springer DE, Berlin, zweite, erweiterte und aktualisierte auflage Auflage, 2007, ISBN 978-3-540-29968-4.