Bachelor's Thesis

submitted in partial fulfillment of the
requirements for the course "Applied Computer Science"

My Title

John Doe

Institute of Computer Science

Bachelor's and Master's Theses
of the Center for Computational Sciences
at the Georg-August-Universität Göttingen

xx. Month 201x

Georg-August-Universität Göttingen
Institute of Computer Science

Goldschmidtstraße 7
37077 Göttingen
Germany

☎   +49 (551) 39-172000
FAX   +49 (551) 39-14403
✉   office@informatik.uni-goettingen.de
🌍   `www.informatik.uni-goettingen.de`

First Supervisor: Prof. Dr. No
Second Supervisor: Prof. Dr. Yes

I hereby declare that I have written this thesis independently without any help from others and without the use of documents or aids other than those stated. I have mentioned all used sources and cited them correctly according to established academic citation rules.

Göttingen, xx. Month 201x

Abstract

Here comes the abstract...

Contents

Chapter 1

Introduction

This is the introduction chapter that shows the basic use of some latex commands that can be used in the document. Of course there is a lot more in latex that cannot be covered here.

Document Structure

The document structure is defined by the `documentclass` command in the main latex file. Here the `book` format is used that supports the following subdivisions:

`\chapter` – A book chapter
`\section` – A section in a chapter
`\subsection` – A subsection in a section of a chapter
`\subsubsection` – A subsubsection in a subsection of section in a chapter

Each of the subdivisions is automatically numbered. If the numbering should be omitted, add a `*` to the command e.g.:

```
\section*{My Section}
```

Additionally, paragraphs can be defined: `\paragraph` – A paragraph
`\subparagraph` – A subparagraph in a paragraph

Paragraph Formatting

Paragraphs can be aligned in different ways:

You can flush a paragraph to the left that is the default.

```
\begin{flushleft}
    You can flush a paragraph to the left that is the default.
\end{flushleft}
```

                                                        Or you can flush a paragraph to the right.

```
\begin{flushright}
    Or you can flush a paragraph to the right.
\end{flushright}
```

                            Or you can center the paragraph.

```
\begin{center}
    Or you can center the paragraph.
\end{center}
```

There is also the `verbatim` paragraph that does not interpret the text. This can be used, for example, to show some source code. However, for an improved source code handling see the `lstlisting` package.

```
#include<stdio.h>

int main() {
    printf("Hello World\n");

    return 0;
}
```

Font

Emphasizing

Text can be *emphasized* as follows:

```
\emph{emphasized text}
```

To emphasize a word **more heavily** do:

```
\textbf{bold text}
```

Font Styles

There are different options to style a font. The most important are:

Use the sans serif font family:

```
\textsf{sans serif font family}
```

Use the teletypefont family (monospace font):

```
\texttt{teletypefont family}
```

Font Sizes

Be aware of the different font sizes:

```
\tiny
```

```
\scriptsize
```

```
\footnotesize
```

```
\small
```

```
\normalsize
```

```
\large
```

```
\Large
```

```
\LARGE
```

```
\huge
```

```
\Huge
```

Colors

The xcolor package allows to set the font color. The easiest way to change the red of a text is to use:

```
\textcolor{red}{red text}
```

There are a couple of pre-defined color names available, but of course custom colors can be defined as well. See the xcolor package manual for details.

List Structures

There are different ways to define a list in latex:

As unordered list:

- One

- Two

- Three

```
\begin{itemize}
    \item One
    \item Two
    \item Three
\end{itemize}
```

Or as ordered list:

1. One

2. Two

3. Three

```
\begin{enumerate}
    \item One
    \item Two
    \item Three
\end{enumerate}
```

Or as description list:

First One

Second Two

Third Three

```
\begin{description}
    \item[First] One
    \item[Second] Two
    \item[Third] Three
\end{description}
```

Of course lists can also be nested:

1. One

    (a) alpha

    (b) beta

2. Two

3. Three

```
\begin{enumerate}
    \item One
        \begin{enumerate}
            \item alpha
            \item beta
        \end{enumerate}
    \item Two
    \item Three
\end{enumerate}
```

Tables

The `tabular` environment can be used to create a simple table:

1	2	3	4
5	6	7	8
9	10	11	12

The (`l`)eft, (`c`)entre, (`r`)ight define the column alignments; the `p` can be used to set a fixed column size (left aligned). The pipe symbol draws a vertical line between the columns. Every column is separated by an `&`, while the rows are separated by `\\`. Horizontal lines can be inserted using the `\hline` command.

```
\begin{tabular}{l|c|p{3cm}|r}
    1 & 2 & 3 & 4 \\
    5 & 6 & 7 & 8\\[6pt]
    \hline
    9 & 10 & 11 & 12 \\
\end{tabular}
```

To span multiple rows or columns see the `multirow` package documentation. For flexible column sizes check the `tabularx` package.

Table Environment

To center the table, to add a caption and to reference to a table use the `table` environment:

By its label the table can be referenced (see Table 1.1).

1	2	3	4
5	6	7	8
9	10	11	12

Table 1.1: This is a Table

The `table` environment has a parameter to help latex to position the table. The provided positioning options will be tried in the given order (h)ere, (t)top, (b)ottom, separate (p)age.

```
\begin{table}[htbp]
    \centering
    \begin{tabular}{l|c|p{3cm}|r}
        1 & 2 & 3 & 4 \\
        5 & 6 & 7 & 8\\[6pt]
        \hline
        9 & 10 & 11 & 12 \\
    \end{tabular}
    \caption{This is a Table}
    \label{tab:firsttable}
\end{table}
```

Using Graphics

The `graphix` package can be used to include graphics in the document. The vector-based graphic formats `eps` and `pdf` are preferred, while the pixel-based `jpg` and `png` can also be included. The extension is automatically figured out by latex:

Different parameters can be provided with the `includegraphic` command such as `height`, `angle`, `scale` etc. However, the most important is the `width` particularly in combination with the percentage of the page with (using the `\textwidth` command).

```
    \includegraphics[width=0.15\textwidth]{images/goe_logo_small}
```

For more details see the `graphicx` manual.

Figure Environment

To center the picture, to add a caption and to reference to the picture use the `figure` environment: The figure can be referenced easily (see Figure 1.1).

Figure 1.1: This is a figure

As for the table environment, the recommended position of the figure can be proposed.

```
\begin{figure}[htbp]
    \centering
    \includegraphics[width=0.15\textwidth]{images/goe_logo_small}
    \caption{This is a figure}
    \label{fig:firstfigure}
\end{figure}
```

Citing

One of the most important things in scientific work is the citing. In latex citing, the corresponding numbering and formatting is easy since everything is automatically managed by latex. For example:

John Doe [1] proposes is his paper a new approach on XYZ.

Meyer et al. [2, p. 100] suggest a new method to XYZ.

The `\cite` command is used to reference to an item by its key. The references are stored in a separate `references.bib` file that contains all references in the `bibtex` format.

```
    \cite{doe2013}
    \cite[p. 100]{meyer2014}
```

The bibliography is automatically created based on the used citations.

For good practices in scientific writing you can make use of the following documents:

- `https://www.hochschulverband.de/fileadmin/redaktion/download/pdf/resolutionen/Gute_wiss._Praxis_Fakultaetentage.pdf` (only in german)

- `https://www.sub.uni-goettingen.de/en/learning-teaching/`
  `academic-work-tools-and-methods/academic-writing/`

Footnotes

Footnotes can be easily inserted using the `footnote` command[1].

`\footnote{Make sure to used footnotes only where really necessary!}`

For manual placement of the mark and the corresponding text use the `\footnotemark` and the `\footnotetext` command.

Mathmode

Latex provides very sophisticated options to display mathematical formulas. The easiest way to enter the math mode within the text is to use `$ ... $` e.g. `$ \alpha + \beta = 12345 $` results in $\alpha + \beta = 12345$

Equations can also be put into a separate environment that can be referenced:

$$f(n) = n^5 + 4n^2 + 2 \tag{1.1}$$

The equation can be referenced (see Equation 1.1).

```
\begin{equation}
    f(n) = n^5 + 4n^2 + 2
    \label{eq:firstequation}
\end{equation}
```

There is a lot more about math in Latex, which cannot be considered here. Please check the corresponding documents.

Listings

As presented before, for simple commands that should not be interpreted the `verbatim` environment can be used. For a more sophisticated syntax highlighting use the `lstlisting` package.

```
#include<stdio.h>
```

---

[1]Make sure to used footnotes only where really necessary!

```
int main() {
    printf("Hello World\n");

    return 0;
}
```

Listing 1.1: Hello World

The listing can also be referenced (see Listing 1.1).

```
\begin{lstlisting}[caption={Hello World},label={lst:helloworld}]
#include<stdio.h>

int main() {
    printf("Hello World\n");

    return 0;
}
\end{lstlisting}
```

To include a whole file use:

```
\lstinputlisting{source_filename.py}
```

For more details about the formatting options and available programming languages consult the `lstlisting` manual.

Acronyms

Using the `acronym` package, it is easy to use acronyms that are automatically listed in the acronym table. The acronyms are located in a separate file, but can be accessed using the keyword.

For example:

FYI! (**FYI!**) is a common abbreviation. **FYI!** can be used in multiple contexts. It is used so often because **FYI!** is very popular.

```
\ac{FYI} is a common abbreviation. \ac{FYI} can be used in multiple contexts.
It is used so often because \ac{FYI} is very popular.
```

Using `\ac` for the first time results in showing the complete form, while every following time just an abbreviation is used. The use of the complete form can be forced by using `\acf`.

PDF/A validation

The thesis has to be in PDF/A format for archiving. This template automatically generates your file according to the PDF/A standard. It can be that this is violated by changing the document, for example by inserting non-compliant graphics. Please check, whether your final version of the document is PDF/A compliant, e.g., by using a freely available tool like *veraPDF*[2].

---

[2]https://verapdf.org/home/

Chapter 2

Basics

In this chapter, the basics of ...

Chapter 3

Analysis

In this chapter, the analysis of ...

Chapter 4

Design

In this chapter, the design of ...

Chapter 5

Implementation

In this chapter, the implementation of ...

Chapter 6

Conclusion

In this chapter, the conclusion of ...

Bibliography

[1] J. Doe, "A very interesting article," *Famous Journal*, vol. 2, no. 3, pp. 10–15, Jun 2013.

[2] B. Meyer, S. Smith, and M. Green, *The Book*.   PublishMe Ltd., May 2014.