



# MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis  
„I wish I had a Master's Degree“

verfasst von / submitted by  
FirstName LastName BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of  
Master of Science (MSc)

Wien, 2019 / Vienna, 2019

Studienkennzahl lt. Studienblatt /  
degree programme code as it appears on  
the student record sheet:

UA 126489

Studienrichtung lt. Studienblatt /  
degree programme as it appears on  
the student record sheet:

Masterstudium Informatik

Betreut von / Supervisor:

Univ. Prof. My favorite professor, PhD

Mitbetreut von / Co-Supervisor:

Professor Co  
Second Row



# Acknowledgements

Thank you!



# Abstract

This L<sup>A</sup>T<sub>E</sub>X template provides example on how to format and display text, mathematical formulas, and insert tables or images. There is a lot more you can do with L<sup>A</sup>T<sub>E</sub>X, for more information check out <https://en.wikibooks.org/wiki/LaTeX>.



# Kurzfassung

Das ist eine deutsche Kurzfassung meiner in Englisch verfassten Masterarbeit.





# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>Kurzfassung</b>	<b>v</b>
<b>List of Tables</b>	<b>ix</b>
<b>List of Figures</b>	<b>xi</b>
<b>List of Algorithms</b>	<b>xiii</b>
<b>Listings</b>	<b>xv</b>
<b>1. Using the template</b>	<b>1</b>
1.1. Using the template . . . . .	1
1.1.1. Title-page . . . . .	2
1.2. Creating the pdf . . . . .	2
1.2.1. Embed all fonts in pdf . . . . .	2
1.2.2. Making a PDF/A-1 compatible pdf . . . . .	3
1.2.3. General Remarks to create the pdf . . . . .	5
1.3. Further tips on the template . . . . .	5
<b>2. Ordinary Text</b>	<b>7</b>
<b>3. Displayed Text</b>	<b>9</b>
<b>4. Tables and Images</b>	<b>11</b>
4.0.1. Tables . . . . .	11
4.0.2. Images . . . . .	11
<b>5. Code</b>	<b>13</b>
<b>6. Algorithms</b>	<b>15</b>
<b>7. Acronyms and Glossary.</b>	<b>17</b>
<b>8. Macros</b>	<b>19</b>
<b>Bibliography</b>	<b>21</b>
<b>A. Appendix</b>	<b>27</b>



# List of Tables

4.1. some description of the table . . . . .	11
--	----



# List of Figures

4.1. Image Example . . . . .	11
------------------------------	----



# List of Algorithms

1. How to write algorithms . . . . .	15
--------------------------------------	----





# Listings

5.1. Python example . . . . .	13
-------------------------------	----



# 1. Using the template

Welcome to the Latex Template of the Computer Science department of the university of Vienna. <https://informatik.univie.ac.at/en/>

General information about finishing your Bachelors degree can be found here:

1. <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/studienabschluss/abschluss-des-bachelor-studiums/>
2. <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/bachelorarbeit-empfehlungen/>

General information about finishing your Masters degree can be found here:

1. <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/wegweiser-masterstudium/approbation-der-masterarbeit/>
2. <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/wegweiser-masterstudium/anmeldung-masterarbeit-themenfindung/>
3. <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/wegweiser-masterstudium/anmeldung-zur-masterpruefung/>

General information about finishing your Doctors degree can be found here:

1. <https://doktorat.univie.ac.at/doktoratsablauf/abschlussphase/einreichen-und-begutachtung/>

## 1.1. Using the template

Everything in the template revolves around the `main.tex` file. Here all the other files are put together to create the thesis. There are 3 different sections to consider: Front Matter, Chapters and Back Matter. Each section has a folder where you can put the different parts of your thesis. In the Front Matter section you should put everything that comes before your first chapter of the thesis. Respectively, in the Back Matter section you put everything that comes after your last chapter. And finally all the chapters are put into the Chapters folder. You can put thing like abstract, summaries, etc... wherever they suit your thesis. In the end it really only matters how you add them into your `main.tex` file. With the `\input{}` command you can add the parts if they should appear in your thesis, the order within the `main.tex` also determines the order in the final pdf. Also depending on whether you do your BSc, MSc or Phd you should add the corresponding titlepage in the `main.tex` file and then delete the tex files of the other two.

## 1. Using the template

### 1.1.1. Title-page

The Layout of the Title-page is in the `UniVieCS_Thesis.cls` and should normally not be changed. You can use all the commands as seen in the `Titlepage.tex`.

The template and also the title-page is based on the `scrbook` class. So if you want to use a different class without changing the title page I would recommend using the template for creating the title page and then use it in your project using the `\includepdf{<filename>}` from the `pdfpages` package. Alternatively you can also always use the word template for the title- page and add it using this package. In that way you can avoid any differences to the original title page template. The word templates can be found here:

1. <https://informatik.univie.ac.at/en/study/support-for-students/bachelor-thesis-guidelines/>
2. <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/wegweiser-masterstudium/approbation-der-masterarbeit/>
3. <https://doktorat.univie.ac.at/doktoratsablauf/abschlussphase/einreichen-und-begutachtung/>

If you use the `includepdf` package make sure that the pdf still has the correct metadata afterwards.

## 1.2. Creating the pdf

To create a proper pdf file of your thesis there are some things to consider

### 1.2.1. Embed all fonts in pdf

Please make sure that you embed all fonts in your pdf. Also make sure all the fonts of any figures that were used in the document are embedded. If you don't use any pdf figures pdfLaTeX should embed all fonts automatically.

#### Linux

On Linux you can use the command

```
pdffonts my_file.pdf
```

to check if the fonts are embedded. Check if all the fonts listed have a "yes" in the "emb" column.

name	type	encoding	emb	sub	uni
BXJBCJ+NimbusSanL-Bold	Type 1	Custom	yes	yes	no
HEMYJL+NimbusSanL-Regu	Type 1	Custom	yes	yes	no
OOJWDR+SFRM1000	Type 1	Custom	yes	yes	no
OHLNOC+SFRM0900	Type 1	Custom	yes	yes	no
...					

For more information see

<https://www.karlsruh.net/2016/01/embed-all-fonts-in-pdfs-latex-pdflatex/>

## Windows

On Windows using the Adobe Acrobat Reader the fonts can be found at

File > Properties > Fonts

For more information please consult

1. <https://helpx.adobe.com/acrobat/using/pdf-fonts.html>
2. [https://www.overleaf.com/learn/latex/Questions/My\\_submission\\_was\\_rejected\\_by\\_the\\_journal\\_because\\_%22Font\\_XYZ\\_is\\_not\\_embedded%22.\\_What\\_can\\_I\\_do%3F](https://www.overleaf.com/learn/latex/Questions/My_submission_was_rejected_by_the_journal_because_%22Font_XYZ_is_not_embedded%22._What_can_I_do%3F)

### 1.2.2. Making a PDF/A-1 compatible pdf

As can be seen in [https://hopla.univie.ac.at/erstellen\\_von\\_pdf.pdf](https://hopla.univie.ac.at/erstellen_von_pdf.pdf) and the "Informationen zur Erstellung und Abgabe von Hochschulschriften" it is required to provide a PDF/A-1 or PDF/A-2 version of your thesis.

#### validation

To validate the produced PDF you can either use the Preflight tool included in Adobe Acrobat Pro or a free online version. E.g. <https://www.pdf-online.com/osa/validate.aspx>. Please take caution as different validation tools can report different results.

#### pdfx

First step to get PDF/A compatibility is by using the package `pdfx` - make sure it is included before the `hyperref` package.

```
\usepackage[a-1b]{pdfx}
```

This package is already included in the class. For more details about the package and the following steps please consult <http://texdoc.net/texmf-dist/doc/latex/pdfx/pdfx.pdf>

#### metadata

There is a section in the beginning of the `.tex` file where you can change the metadata. Change your Name, Title, Subject, Keywords and remove/add Information as you like. To find out what fields are possible please check here. <http://texdoc.net/texmf-dist/doc/latex/pdfx/pdfx.pdf#subsection.2.3>

A file containing the metadata (`jobname.xmpdata`) is created when compiling.

Watch out, the metadata `.xmpdata` file is only created one time. So if you need to update it you need to clear the cache of overleaf. Or delete the `.xmpdata` file. It is then recreated the next time you compile.

[https://www.overleaf.com/learn/how-to/Clearing\\_the\\_cache](https://www.overleaf.com/learn/how-to/Clearing_the_cache)

You can check the metadata of your PDF with Acrobat Reader by going to File->properties. Or alternatively check it with an online tool.

## 1. Using the template

### figures

As already mentioned, make sure that all the fonts used in the pictures are included. Furthermore transparency in pictures causes issues, please convert transparent figures into their nontransparent version.

Using Linux the command `pdfimages -list <pdf>` shows the type of all images used. The type should always be `image` and not `smask`. Check and convert these images.

### color

Additionally there can be problems if figures use different color spaces. Use the same command as before and check if all images use the same color.

If color is really important in your work it might also be a good idea to use an ICC profile for the color. For more details about colors check <http://texdoc.net/texmf-dist/doc/latex/pdftex/pdftex.pdf#subsection.2.5>

It is also possible to convert the pictures automatically using ghostscript. But always check the results manually.

### Other errors

Due to the complexity of Latex files there can be many more errors that are not covered in the readme.

The Preflight tool included in Adobe Acrobat Pro also has the ability to fix some errors. For example EOL (End of Line) errors can be fixed with its analyze and fix option. Please also check if any of the following pages might have a solution to your problem:

1. <https://www.mathstat.dal.ca/~selinger/pdfa/>
2. <https://blog.zhaw.ch/icclab/creating-pdfa-documents-for-long-term-archiving/>
3. german: <http://kulturreste.blogspot.com/2014/06/grrrr-oder-wie-man-mit-latex-vielleicht.html>
4. [https://support.stmdocs.in/wiki/?title=Generating\\_PDF/A\\_compliant\\_PDF\\_s\\_from\\_pdftex](https://support.stmdocs.in/wiki/?title=Generating_PDF/A_compliant_PDF_s_from_pdftex)
5. <http://texdoc.net/texmf-dist/doc/latex/pdftex/pdftex.pdf>

### Tagged PDF

Currently with Latex it is only possible to create files that are in the PDF/A-1b format. The PDF/A-1a format required the PDF to be tagged which is currently not possible in a satisfactory way. A manual tagging with Adobe Acrobat Pro is possible but not recommended.

More information about the current status of tagged pdfs can be found here:

1. <https://umij.wordpress.com/2016/08/11/the-sad-state-of-pdf-accessibility-of-latex-documents/>
2. <https://www.tug.org/TUGboat/tb30-2/tb95moore.pdf>

### 1.2.3. General Remarks to create the pdf

Highest priority should always be the embedding of all fonts. Further compliance with the PDF/A standards is always desired, but talk to your supervisor in any case.

Please also check the following resources if you have problems and need assistance

1. [https://spl29.univie.ac.at/fileadmin/user\\_upload/s\\_spl29/Studium/abschluss\\_master/Infoblatt\\_Hochschulschriften.pdf](https://spl29.univie.ac.at/fileadmin/user_upload/s_spl29/Studium/abschluss_master/Infoblatt_Hochschulschriften.pdf)
2. [https://e-theses.univie.ac.at/E-Theses\\_erstellen\\_von\\_pdf.pdf](https://e-theses.univie.ac.at/E-Theses_erstellen_von_pdf.pdf)

### 1.3. Further tips on the template

In the following chapters there are some general tips on the elements of Latex. You can check them out if you think they have useful information for you. Then you delete these chapters and replace them with the real chapters of your thesis.

Also don't forget to register your Thesis at: <https://informatik.univie.ac.at/studium/hilfe-fuer-studierende/wegweiser-masterstudium/anmeldung-masterarbeit-themenfindung/>





## 2. Ordinary Text

The ends of words and sentences are marked by spaces. It doesn't matter how many spaces you type; one is as good as 100. The end of a line counts as a space.

One or more blank lines denote the end of a paragraph.

Since any number of consecutive spaces are treated like a single one, the formatting of the input file makes no difference to  $\text{\LaTeX}$ , but it makes a difference to you. When you use  $\text{\LaTeX}$ [Lam94], making your input file as easy to read as possible will be a great help as you write your document and when you change it. This sample file shows how you can add comments to your own input file.

Because printing is different from typewriting, there are a number of things that you have to do differently when preparing an input file than if you were just typing the document directly. Quotation marks like “this” have to be handled specially, as do quotes within quotes: “ ‘this’ is what I just wrote, not ‘that’ ”.

Dashes come in three sizes: an intra-word dash, a medium dash for number ranges like 1–2, and a punctuation dash—like this.

A sentence-ending space should be larger than the space between words within a sentence. You sometimes have to type special commands in conjunction with punctuation characters to get this right, as in the following sentence. Gnats, gnus, etc. all begin with G. You should check the spaces after periods when reading your output to make sure you haven't forgotten any special cases. Generating an ellipsis . . . with the right spacing around the periods requires a special command.

$\text{\LaTeX}$  interprets some common characters as commands, so you must type special commands to generate them. These characters include the following: \$ & % # { and }.

In printing, text is usually emphasized with an *italic* type style.

*A long segment of text can also be emphasized in this way. Text within such a segment can be given additional emphasis.*

It is sometimes necessary to prevent  $\text{\LaTeX}$  from breaking a line where it might otherwise do so. This may be at a space, as between the “Mr.” and “Jones” in “Mr. Jones”, or within a word—especially when the word is a symbol like *itemnum* that makes little sense when hyphenated across lines.

Footnotes<sup>1</sup> pose no problem.

$\text{\LaTeX}$  is good at typesetting mathematical formulas like  $x - 3y + z = 7$  or  $a_1 > x^{2n} + y^{2n} > x'$  or  $AB = \sum_i a_i b_i$ . The spaces you type in a formula are ignored. Remember that a letter like  $x$  is a formula when it denotes a mathematical symbol, and it should be typed as one. Furthermore you can add a formula as Images or Tables, see Formula 2.1

$$a + b = c \tag{2.1}$$

It is sometimes necessary to prevent  $\text{\LaTeX}$  from breaking a line where it might otherwise do so. This may be at a space, as between the “Mr.” and “Jones” in “Mr. Jones”, or within

---

<sup>1</sup>This is an example of a footnote.

## 2. Ordinary Text

a word—especially when the word is a symbol like *itemnum* that makes little sense when hyphenated across lines.

### 3. Displayed Text

Text is displayed by indenting it from the left margin. Quotations are commonly displayed. There are short quotations

This is a short quotation. It consists of a single paragraph of text. See how it is formatted.

and longer ones.

This is a longer quotation. It consists of two paragraphs of text, neither of which are particularly interesting.

This is the second paragraph of the quotation. It is just as dull as the first paragraph.

Another frequently-displayed structure is a list. The following is an example of an *itemized* list.

- This is the first item of an itemized list. Each item in the list is marked with a “tick”. You don’t have to worry about what kind of tick mark is used.
- This is the second item of the list. It contains another list nested inside it. The inner list is an *enumerated* list.
  1. This is the first item of an enumerated list that is nested within the itemized list.
  2. This is the second item of the inner list.  $\text{\LaTeX}$  allows you to nest lists deeper than you really should.

This is the rest of the second item of the outer list. It is no more interesting than any other part of the item.

- This is the third item of the list.

You can even display poetry.

There is an environment for verse  
Whose features some poets will curse.

For instead of making  
Them do *all* line breaking,

It allows them to put too many words on a line when they’d rather be forced  
to be terse.

Mathematical formulas may also be displayed. A displayed formula is one-line long; multiline formulas require special formatting instructions.

$$\Gamma \times \psi = x'' + y^2 + z_i^n$$

Don’t start a paragraph with a displayed equation, nor make one a paragraph by itself.



## 4. Tables and Images

One of the great advantages of L<sup>A</sup>T<sub>E</sub>X is that all it needs to know is the structure of a document, and then it will take care of the layout and presentation itself. So, here we shall begin looking at how exactly you tell L<sup>A</sup>T<sub>E</sub>X what it needs to know about your document.

### 4.0.1. Tables

In this sub-section, a simple table is inserted. To add reference to the table, see (cf. Table 4.1):

Command	Level
<code>\part{part}</code>	-1
<code>\chapter{chapter}</code>	0
<code>\section{section}</code>	1
<code>\subsection{subsection}</code>	2
<code>\subsubsection{subsubsection}</code>	3
<code>\paragraph{paragraph}</code>	4
<code>\subparagraph{subparagraph}</code>	5

Table 4.1.: some description of the table

### 4.0.2. Images



Figure 4.1.: Image Example

When an image is inserted, you can refer to it like this (cf. Figure 4.1).

### A Subsubsection

As one last example, this is how you can insert a sub-sub-section! Have fun writing your thesis with L<sup>A</sup>T<sub>E</sub>X!



## 5. Code

If you want to show program code within your thesis you can use the `\texttt{verbatim}` environment or for a more complex display take a look at [https://www.overleaf.com/learn/latex/Code\\_listing](https://www.overleaf.com/learn/latex/Code_listing)

Text enclosed inside `\texttt{verbatim}` environment is printed directly and all `\LaTeX{}` commands are ignored.

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M
```

Listing 5.1: Python example





## 6. Algorithms

If you want to show algorithms in your Thesis take a look at the <https://www.overleaf.com/learn/latex/algorithms> page. The `algorithm2e` package is already included in the template. You can list algorithms in the same way as you can list Tables and Figures.

**Data:** this text

**Result:** how to write algorithm with  $\text{\LaTeX}2\text{e}$  initialization;

```
while not at end of this document do  
  read current;  
  if understand then  
    go to next section;  
    current section becomes this one;  
  else  
    go back to the beginning of current section;  
  end  
end
```

**Algorithm 1:** How to write algorithms



## 7. Acronyms and Glossary.

if you want to use Acronyms or a Glossary check the page here: <https://www.overleaf.com/learn/latex/glossaries>

The Latex typesetting markup language is specially suitable for documents that include mathematics. are rendered properly an easily once one gets used to the commands.

Given a set of numbers, there are elementary methods to compute its Greatest Common Divisor, which is abbreviated GCD. This process is similar to that used for the Least Common Multiple (LCM).



## 8. Macros

You can also add useful packages or macros into the `packages_macros.tex` file to add them to the project. The packages for algorithms, code or the glossary have already been added there.

### FixMe

Another example, the `FixMe` package, is added as well. It allows you or your supervisor to add Meta comments to the document. These comments only appear if you set the draft mode in the `main.tex` file. If you remove or comment the activation of the draft mode you can see your final thesis without comments. *This line can only be seen if the draft mode is set in the main.tex – `FixMe`*

! – *FixMe*



# Bibliography

- [Lam94] Leslie Lamport. *LaTeX – A Document Preparation System*. Addison-Wesley, second edition, 1994. User’s Guide and Reference Manual.





# Acronyms

**GCD** Greatest Common Divisor. 17

**LCM** Least Common Multiple. 17



# Glossary

**latex** Is a mark up language specially suited for scientific documents. 17

**mathematics** Mathematics is what mathematicians do. 17



## A. Appendix

here you can put further things you want to add like transcripts, questionnaires, raw data...