BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEVELOPMENT OF AN LLM RED-TEAMING TOOLKIT
VÝVOJ TOOLKITU PRO RED-TEAMING VELKÝCH JAZYKOVÝCH MODELŮ (LLM)

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTHOR   **ADAM VESELÝ**
AUTOR PRÁCE

SUPERVISOR   **Ing. JAKUB REŠ**
VEDOUCÍ PRÁCE

BRNO 2026

Abstract

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v anglickém jazyce.

Abstrakt

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v českém (slovenském) jazyce.

Keywords

Sem budou zapsána jednotlivá klíčová slova v anglickém jazyce, oddělená čárkami.

Klíčová slova

Sem budou zapsána jednotlivá klíčová slova v českém (slovenském) jazyce, oddělená čárkami.

Reference

VESELÝ, Adam. *Development of an LLM Red-Teaming Toolkit*. Brno, 2026. Bachelor's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Ing. Jakub Reš,

Development of an LLM Red-Teaming Toolkit

Declaration

I hereby declare that this Bachelor's thesis was prepared as an original work by the author under the supervision of Mr. Ing. Jakub Reš. I have listed all the literary sources, publications and other sources, which were used during the preparation of this thesis. I have used ChatGPT to correct spelling and other language mistakes. I used Grok when working on the software.

<div align="right">

. . . .
Adam Veselý
November 24, 2025

</div>

Acknowledgements

I would like to thank my supervisor Ing. Jakub Reš for his guidance and support throughout the development of this thesis. I also appreciate the assistance provided by my colleagues and friends who contributed their insights and expertise.

Contents

List of Figures

Chapter 1

Introduction

The rapid advancement and widespread deployment of large language models (LLMs) such as GPT-5.1, Claude 4.5, Gemini 3 and others have transformed natural-language interaction with computers. These models power chatbots, code assistants, translation services, and creative tools used daily by millions of users.

However, their remarkable capabilities come with significant safety and ethical risks. LLMs can generate harmful, biased, misleading, or illegal content when subjected to carefully crafted adversarial inputs, a practice commonly known as *jailbreaking* [7], [8].

Real-world incidents such as ChatGPT being tricked into providing bomb-making instructions [2], or Gemini's image-generation controversy, have demonstrated that even heavily aligned commercial models remain vulnerable.

Red teaming, cybersecurity technique involving simulated attacks to expose vulnerabilities, has been adopted by leading AI organisations (OpenAI, Anthropic, Google DeepMind, Meta AI) as a core component of LLM safety evaluation [3].

With the adoption of the EU AI Act in 2024, systematic risk assessment including red teaming will become a legal requirement for high-risk AI systems deployed in the European Union from 2026 onward [4].

Consequently, efficient, reproducible, and extensible red-teaming tools are no longer a luxury but an essential part of responsible AI development.

Despite significant progress, most existing open-source red-teaming frameworks suffer from limited modularity, poor support for multi-turn conversations, inadequate multilingual harm detection, or steep learning curves that hinder adoption by smaller research teams and individual developers [5, 6, 1].

This creates a clear need for a new, developer-friendly toolkit that lowers the barrier to systematic safety testing.

The main goal of this bachelor's thesis is therefore the design, implementation, and evaluation of a modular open-source red-teaming toolkit for large language models that addresses the identified shortcomings of current solutions.

The specific objectives assigned for this work are:

1. To study the current state of research in red-teaming of large language models, including attack tactics (jailbreaking, prompt injection, multi-turn adversarial prompts), threat types, and contemporary safety evaluation approaches.

2. To perform a survey of existing open-source tools and frameworks for LLM red-teaming (e.g. PyRIT, Garak, MAD-MAX), analyse their architectures, advantages, and limitations.

3. To propose a modular architecture of a new red-teaming toolkit that enables detection of LLM weaknesses, definition of various attack types, automation of testing scenarios, and structured result evaluation.

4. To implement a functional prototype supporting both manual and automated attack generation and evaluation on selected models (e.g. Llama 3, GPT-OSS).

5. To test the system's functionality, focusing on attack success rate and robustness of the testing environment.

The present document submitted in the winter semester 2025/2026 covers objectives 1–3 (literature review, survey of existing solutions, and architectural design).

The implementation and experimental evaluation (objectives 4–5) will be completed in the summer semester 2026.

The thesis is structured as follows: Chapter **??** provides background on LLM safety risks and red-teaming methodology.

Chapter **??** surveys and compares current open-source red-teaming frameworks.

Chapter **??** presents the proposed modular architecture and its key components.

The implementation, experimental evaluation, and overall conclusions will be added in the final version of this thesis.

Chapter 2

Abstract

First and foremost - abstract matters. Second - It's not that hard to write one. Without further ado, let's dive into it.

2.1 What is the purpose of an abstract

An abstract is used for **searching** purposes, together with the title of the thesis and a list of keywords. These parts (perhaps except for the title) are not directly part of the text and it's not expected that anyone who will read your thesis actually reads them. The fact that they're reading your thesis means they're past the abstract stage. Abstract serves them well to decide **whether or not** they want to read your thesis.

When someone looks for an answer to their problem, they give the librarian or a search engine (these days) keywords that directly relate to their problem. They then receive a list of theses, that could possibly offer a solution based on the match between the keywords used and keywords in the theses. A good thesis title can help the person guess which texts could have a direct relation to their problem and can get them to read your thesis.

This is where the abstract is crucial. The reader reads abstracts of the theses and decides, whether or not they want to read them. It also informs them that their filter based on a title alone is wrong.

At this point, they don't have a PDF with full text or a printed version of the thesis available. Abstracts are **not** supposed to be in the text itself, but to be available on servers aggregating scientific texts. Therefore the first rule is: an abstract needs to work on its own – if it contains references to literature or text ("The efficiency of a method is summarized on page 51."), it only makes a reader less interested in the author, won't read their work or cite them.

2.2 When and how to write an abstract

It makes sense to write an abstract when the writing is done – as a summary and real annotation of the thesis. I however like the opposite approach – write an abstract in the beginning. Whenever I write a scientific article, I start with a long list of keywords that are related to each other. It's a lot more than end up as the final keywords used for indexing. It helps me understand where the article is headed at all times – what should I talk about, what needs to be in the text, what does it deal with? As soon as I'm done with keywords, I form a title and an abstract.

I consider the following four parts of an abstract especially useful – Which problem does it solve? What solution does it offer? What are the results? What is the meaning of these results? Once all of this is clear, the text essentially writes itself. If this is unclear, how on earth can you form a coherent, meaningful sentence in the same text?

2.3   Recommended structure of the abstract

An abstract of a scientific thesis can consist of four parts and be useful. Each part consists of two to three sentences, in some cases even a single sentence is enough.

The term "elevator pitch" is often used in business. It is not a coincidence that its structure is similar to the recommended structure of an abstract. Realistically, an abstract should contain anything the author would say about their scientific thesis if they had at most 2 minutes and could not use slides, images, or text. What should they talk about then?

Part one – What is the problem? What is the topic? What's the goal of the text?

- This thesis deals with.

- The goal of this thesis is.

- My aim was.

There is no place for fairy tales specific to wrong scientific literature: "Our five-year-plan of work open new and bold goals for us", "With the evolution of computing technology and especially the display devices, it is more important than ever . . . " do not belong anywhere near a good text, especially an abstract. If you can express the purpose of your text in one sentence, do it and forget about everything else. Less is always more when it comes to the abstract.

Part two – How is the problem solved? Is the goal fulfilled?

- I solved the problem using this and that.

- I used this method, this procedure and analyzed this.

- The work represents an algorithm that.

- I used these tools to process data and evaluated results like this.

- The principle of our algorithm is.

There is a new methodology in the nature of scientific text (= "how to do something"), it needs to include a description. If the solution consists of three parts, it probably means that this part of an abstract will have three sentences, where each sentence is about a different part of the solution. A good abstract should be honest and accurate in this section – no "revealing secrets" in the text itself. Vague formulations of a solution principle in an abstract usually means that the authors can't write or don't have anything to write about – neither one is good enough to waste your time.

Part three – What are the results? How good is the solution?

- It was 87,3 % successful.

- We created a system that.

- The solution offers these options.

- As a result, we found out that.

Stating a specific number is not a bad habit in this part – "we made existing XY method five times faster". If the contribution of your work cannot be summarized in two or three sentences, something is wrong and the entire text is probably not worth writing.

Part four – Well then? What does it bring to science and the reader?

- The contribution of this thesis is.

- The primary discovery is.

- The primary result is.

- Based on the data it is possible to.

- The results allow us to.

When writing scientific articles, I myself struggle with the similarity of third and fourth part. Both of them speak about the results and contribution of the text. The goal of the third part is to be specific and name achieved results whereas the goal of the fourth part is to interpret their meaning and significance. I guess it's fine if these two statements merge to an extent and both parts not only don't have their own paragraph, but they sometimes even intertwine with their common sentences.

Part zero – What is it about? Where are we?

- The context is this and this.

- It deals with studies of this and that.

- We build on these recent advances in our field.

Sometimes it is necessary to insert a short specification of context at the beginning of your abstract. It can be a great asset when it comes to obscure and esoteric field, that is off to the side of the main flow. Usually, this part is not needed and sentences contained here are prime examples of pseudoscientific nonsense. It's not necessarily bad to forget that this part can even exist in an abstract. If an expert in the field shakes their head after reading an abstract: "I have no idea what this is about." only then it makes sense to include this part to specify context.

2.4   Innovation is not ignorance

In this text, I describe a general model of a general thesis. I would like to state that this is my opinion and taste and I'm interested in alternate opinions and tastes (I really am!). Every graduate (Mgr. and Bc.) feels that their thesis is special and extraordinary. Therefore they won't follow some scheme meant for common and average theses – i.e. the others. I see good reasons to divert from the outlined scheme and recommend some students to divert from the scheme myself every year. Indeed, every thesis is unique and extraordinary. If they weren't, there would be no reason to write them, just copy them instead. Before you divert from the standard and canonical way of organizing scientific text, put some effort into learning, understanding and tackling it. The way of scientific work, structuring scientific text, or citing sources can look rigid and clumsy, but for now it is the best way mankind could come up with. If you learn it, understand its advantages and disadvantages and innovate it, it's great and you're welcome to do so. If you choose to ignore it, you'll most likely end up with a very poor innovation.

Chapter 3

Drafting the basic thesis structure

This chapter contains a selection of useful hints and procedures useful for writing a dissertation (bachelor's thesis is technically a dissertation – you receive a diploma for it) from experienced supervisors. First a number of general principles are listed, followed by a more detailed description of the advised procedure of drafting a thesis structure.

Before you into it head first, ask your supervisor for any advice or if they have their own web page with hints and guidelines. Their area of expertise will probably coincide with that of your thesis and help you with the most appropriate structure that you should comply with. If the authors learn about another collection of useful hints, you'll find it here in the future.

This text deals with the general recommendations and thesis structure, which always has to be modified and thought of based on the specification [**?**].

3.1 Useful hints for writing a technical text

The following instructions can also be found on faculty web pages[**?**]. The overview of basic typography and the creation of documents using LaTeX system can be found in Jiří Rybička's book[**?**].

One of the evaluated parts of a potential engineer is quality of language and literacy. Your goal is to create a clear and comprehensive text. You should express yourself accurately, use the appropriate level of Czech or Slovak grammar (or English if need be) and comply with the generally accepted customs. Slang words and phrases are not allowed. If you are uncertain of the translation or transcription of foreign terms, use literature available in FIT library.

The text should be a short path to understanding a problem, predict it's difficulties and preventing them. Good writing means perfect grammar, correct punctuation and use of appropriate words. You should strive for a good text, one that is not monotonous due to use of small selection of words, or overuse of some of your favorite words. If you use foreign words, it is expected that you know their exact meaning. Obviously, you need to use english words correctly as well. For example, there are certain rules when using the word *obvious*. Is *obvious* really obvious? And did you make sure, that the *obvious* is really valid? You should be careful about using the subject *it* with a passive voice too often. For example, you should never use the Czech *it has proved itself that...*, ever.

It is advised to think the use of symbols for *labeling* through thoroughly. By this we mean a well-thought-out choice of abbreviations and symbols used for distinguishing types

of parts, labeling program's main functions, naming control keys on the keyboard, naming variables in math formulas, etc. Apt and consistent labeling can help readers a lot. It is advised to provide a list of labeling at the beginning of the text. Not just in labeling, consistency is important in references and in typesetting in general.

There are numerous typographical principles that can be used to distinguish things. You can choose different styles for different purposes. As an example, keys can be placed in a rectangle, identifiers from source text can be written in `typewriter font` etc.

Whenever you state facts, you should also state their origin and your attitude to them. If you claim something, you always have to explicitly state, which part of it was proven, what will be proven in your text and what you took from literature (including its source). You should never let the reader doubt whose idea they're presented with, your own or someone else's.

If you want to write a clear and comprehensive academic text, you need to follow these rules:

- You must have something to say,

- you must know, who you want to say it to,

- you must thoroughly think through the contents,

- you must write in a structured way

3.1.1   You must have something to say

The most important prerequisite of a good academic text is to have an idea. If the idea is significant enough, it will last, even if it is not formulated in the best way. However, if you want to articulate your idea as precisely as possible and save the reader's time, you must comply with certain principles discussed further below.

3.1.2   You must know, who you want to say it to

Another important prerequisite of a good academic text is the audience. If you write down notes for yourself, you usually write in a differently than when you write a scientific report, article, thesis, book or a letter. Depending on who the target audience is, you can decide the writing style, the amount of information and how detailed it is.

3.1.3   You must thoroughly think through the contents

You must thoroughly think through the contents of your thesis as well as the order in which you want to present the reader with your ideas. As soon as you know what you want to say and to whom, you need to create a structure. The ideal structure should be one that is logically accurate and psychologically digestible, where everything has its place and its individual parts fit into each other. All the connections between them are clear and it's apparently what belongs where.

To achieve this goal, you must precisely structure the contents of your text. Decide what the main chapters and subchapters are going to be, as well as the connection between them. A diagram of such structure would be a tree-like graph, not a chain. When structuring the contents, it is important to ask yourself what you want to include as well as what you want to exclude. Too much detail could discourage the reader, and so could lack of detail.

The result of this stage is an outline of the text consisting of the main ideas and all the details between connecting them to each other.

3.1.4   You must write in a structured way

You must write in a structured way and work the most comprehensible format simultaneously, with good writing and perfect labelling included. If you have an idea, you know who your target audience is, you know your goal and outline of the text, then you're ready to start writing. When writing your initial draft, you should try to include all your ideas and opinions regarding the individual chapters and subchapters. You need to explain every single idea, describe and prove it. The main idea should always be formulated in a main clause, rather than the dependant clause.

You should approach the writing itself in a structured way too. As you're working on the structure of your thesis, you're creating a framework that you are gradually completing. You should use a DPT[1] program that offers a structured layout of the text (pre-defined types of headings and text blocks).

3.1.5   It will never be perfect

Once you have written everything you thought about, you should take a day or two off, then read what you wrote again. Make the last changes and move on. You should know that something will always remain unfinished and that there is always a better way of explaining something, but every stage of the editing must come to an end.

3.2   Who is the target audience of a thesis

This subsection was taken from professor Herout's blog [**?**].

Write your thesis for a student, that will build on your work.

Imagine that another student will continue the work in your thesis, they're about as smart as you are. You have four hours to show them your thesis, explain everything necessary for them to be able to continue your work. The student studies at the same school and knows about as much as an average student would, they're not an expert in the area of your thesis, but they're not stupid either. The student just found out that they'll continue your work, so they had no time to learn about the topic, just like you.

It's best to start by telling the student what the goal of this thesis is and what the results should look like.

Nobody in their right mind will spend an hour explaining things like this to the student: "Interet was invented by the American army in 1962, then www was invented in CERN in 1991, nowadays it is used for various purposes."(all of this on six pages with countless references and figures). The student usually doesn't need countless pages of details regarding colorful modules to represent figures, history and details of Hough transformation, complete description of the ISO/OSI reference model layers or a set of pie graphs representing individual mobile platforms on the market over the last decade. The student needs to be shown the valuable sources that helped you and wants a brief description of tools and algorithms that were a vital part of the solution: "You need XY tool that does this and

---

[1]Desktop publishing (DTP) - creating printed document on a computer.

that, especially it's PQ module, that is used then and then. The most useful information can be found in this documentation."

Tell the student everything about what worked for you and what helped, but also make sure to tell them what seemed like a good idea, but ended up being useless.

Try to provide just the right amount of detail. Explain an optimization method one line of code at a time, a module in one paragraph enriched with description of input and output data, and a reference to the respective library.

Imagine this four hours long session with the student.

- What do you think you would talk about in the beginning, how long did it take for the student to understand?

- What are some things that should be mentioned?

- What kind of figures would you draw during the session?

- What would the student ask about, what is important but not clear?

- What restrictions and unfinished things would you need to inform the student about to prevent them from falling into a trap?

- How can the student continue? What is left unfinished and is worth trying? What could be improved?

- What would you say in the very first and very last minute of the session?

3.3   Thesis structure − Five chapters

This subsection was taken from the blog of professor Herout[**?**] (partially inspired by a book from Jean-Luc Lebrun [**?**]) and from a document on professor Zemčík's web page [**?**].

Thesis is something that students work on for 2 semesters of their studies and then write a small book about it. The misconception is that this little book is the master's thesis. The book is in fact a technical report about the year long work and master's thesis represents the result of student's work.

The year long work includes studies first and foremost: "What already exists in the area of my thesis? How did the others do it?" It is implied that a student tries to innovate and design some things: "The problem has several solutions, I chose this approach, because it is the most efficient option for the given platform." The researcher should implement and evaluate their designs to validate them: "I used these tools for implementation, the entire system is split into these modules. The result is this fast, it is this effective and user reviews are such and such."

The basic structure of master's thesis according to professor Herout is as follows:

1. Introduction (1 page)

2. What had to be studied (including assessment of the current state; 40 %)

3. New ideas that this thesis explores (30 %)

4. Implementation and evaluation (30 %)

5. Conclusion (1 page)

If the text has these 5 chapters, it is not a mistake, neither is if one of them is split in two parts – more on that later. A big mistake is when any of the parts are missing, or when it's content differs from the rest.

Names of chapters can vary from this structure, even though your thesis will have an introduction and a conclusion. The content of your thesis itself is what really matters and if it means you have to break the structure, then do so.

Many supervisors agree with this basic structure, even though some of them recommend different chapter titles for example assessment of the current state can be used in the second chapter, and even in the third chapter according to professor Zemčík:

1. Introduction (1–2 pages)

2. Summary of the current state (40–50 %)

3. Assessment of the current state and solution draft (3-5 pages)

4. Your own work (roughly 40 %)

5. Conclusion (at most 1 page)

Opinions of supervisors also differ in length depending on the specification of thesis, this can be seen for example in recommendations from assistant professor Beran [**?**]:

1. Introduction (1 page)

2. Theory (1/3 of pages)

3. Solution draft (1/3 of pages)

4. Implementation, experiments and assessment (1/3 of pages)

5. Conclusion (1 page)

When it comes to practice-oriented theses, where the most important things are data and user interface, associate professor Černocký [**?**] recommends the following:

1. Introduction (single digit of pages)

2. Theoretical part (roughly 10 pages)

3. Data (single digit of pages)

4. Breakdown of Your algorithm and testing (roughly 10 pages)

5. Draft and implementation (several pages)

6. User interface (several pages)

7. Testing (roughly 10 pages)

8. Conclusion (single digit of pages)

3.4 Thesis – comics edition

This subsection was taken from professor Herout's blog. [**?**].

Thesis (including bachelor's) is a fairly complex work comprised of a large number of letters. These letters follow a certain hierarchy, organised into chapters. The whole thing should follow a logical order – reader should first learn one thing so that they can then learn and understand other things. It should contain figures, tables, formulas; these non-textual objects should fit in with the surrounding text and enrich it. Every single aspect of the specification of your thesis has to be explored, it has to be finished by a certain date, printed and bundled into a book. If you want your thesis to be good, you need to make it good. Whenever you bake a cake from ten ingredients and one of them is rotten, it doesn't matter how hard you try, the cake will not taste good.

3.4.1 How should you approach everything?

Whenever we write an article (all the time lately), we create something called a "Comics Edition" in the early stage. We do it because I insist that we do it and because it helps us a lot. Perhaps it can help you with your thesis.

First, make sure you know the answers to the following questions:

- How would you explain the principle of your solution in three to five short sentences?

- What are the strengths of your solution?

- What arguments would you use to support the correctness of your solution?

- If someone wanted to criticize your thesis – what would they criticize it for?

- What would your answer be?

- What keywords should one use in a search engine to find your thesis in the search results?

If you are finished, let's get into it...

3.4.2 Create THE document

I often see people write an "initial" version of a thesis somewhere in their notes. First of all, that's extra work and second it is entirely pointless. It's best to create a document where the fight begins and also ends.

3.4.3 Chapter titles

Chapter titles are an important part of a comics edition, put them in your document.

Put them in for the sake of formatting – no provisional bullet point lists: "I'll just do it later". You need to see how the automatically generated content looks now, before it's too late to change everything.

Put them in for the sake of wording. Chapter title says, what the chapter is about. Chapter titles represent a skeleton covered in flesh and skin of text and figures.

Of all the words in your thesis, words in the titles are the **m**ost important ones. Put some time and effort into your titles.

3.4.4 Figures

A picture is worth a thousand words. I went through the last 8 articles that I helped write. They're 80 pages total and contain 87 figures and 17 tables, that's 1,3 of visual information per page (including pages containing references, title pages with abstracts etc). Many figures (about half) are comprised of multiple subfigures, especially referenced figures. I counted 221 subfigures in the mentioned articles, that's an average of 3 visuals per page. This is what I imagine the role of figures in a scientific text should be. I think a thesis with "too many figures" does not exist.

You should think about what images you will use and where they'll be even in the early stage of comics edition.

Figures don't have to be finished. We don't know what exactly the figure will represent just yet. We don't know what the caption will be. We only know that there will be a figure and that it will be comprised of multiple subfigures. It takes roughly a minute (we already have a vector format of a "TODO Image") and it shows us how the text will look.

In some cases, we know what the images will look like on a conceptual level. Draw it on a paper or a blackboard, take a picture of it, and insert it where the final figure will be (vector image made in Inkscape[2] or generated using Gnuplot[3]).

As a side note: A picture is worth a thousand words. Stupid picture is worth a thousand stupid words.

Just one more thing on the side when it comes to figures: If instead of vector images (schemes, graphs, drafts, diagrams, essentially everything except photos and screenshots) you use bitmap images and you use screenshots with a lossy compression (usually JPEG), don't expect a positive feedback or assessment.

3.4.5 Quantity of text

Just like with figures we don't have yet, we insert even text that we don't have yet.

LaTeX has a beautiful command \Blindtext[4] just for that. If you don't want to use it or don't know how, use http://lipsum.com. It helps you estimate the length of your work, figure density in text and other characteristics. It takes roughly 5 minutes to create this sort of estimate. To make sure you don't get lost in your work-in-progress text, change its color to grey (a smart person creates a command to change the color of the text unanimously for the whole document). From experience, it's easy to get lost without colors – what is finished, what isn't and what needs to be worked on. It is advised to spend a few minutes getting a text coloring package to work. You can use command \todo in this template, for example [[**This needs to be finished**]].

The genesis of each chapter begins with 3-5 TODO pieces and some Lorem ipsum. Each TODO is then transformed into a larger number of TODOs of a smaller scale or into text, figure, other subchapters, and pretty much anything. TODOs come and go, but they always move you closer to the final product.

---

[2]https://inkscape.org
[3]http://www.gnuplot.info/
[4]Short tutorial: https://texblog.org/2011/02/26/generating-dummy-textblindtext-with-latex-for-testing/

3.4.6 How do you work with it

When you sit down and LaTeX is having a good day, it takes you an hour to write a thesis (bachelor's thesis included), with the right number of pages and a pretty good idea about where everything will be. It is slowly forming the result that is yet to come, after some more work.

The document does not really grow in size anymore, but it transforms. There's a big difference between sitting down in front of a blank page and "write a thesis" and transforming one TODO into a paragraph. The first one is difficult and sometimes you just can't make it work. The latter works: it has its head and tail. At least you know what to do.

Still, the thesis won't write itself, but it gets a lot easier and the final result is that much better.

3.5 Chapter titles

To publish well does not mean to fill up as many pages with letters as possible. Scientist's achieve their renown when their work is useful enough to other scientists, that they end up citing them. Therefore it is important that one's article is well written: nobody will cite a thesis that is poorly written because it degrades them.

Though poorly written article won't be cited because **nobody can find it**. Long before the internet SEO even existed, scientists used all kinds of methods to make sure that other scientists working on their recherche will include others' materials that they read, make notes and finally – cite in their work.

There are a lot of articles in the world. Whenever a scientist searches for materials relevant to their work, they enter keywords (formerly on paper in library, nowadays electronically into a search engine) and they get a lot of results – e.g. titles. First step to get someone to cite an article is to have a good title. Title so good that a scientist is interested in your article enough to actually read it and find out more. Title is the **first filter**.

The scientist then opens the articles that satisfy the requirements of first filter. That means they get to see the abstract of the article. Abstract is the **second filter** and quite important one. It's similar to getting to a second round of interview for your dream job. People tend to care at that point.

If the scientist is interested in the title and the abstract, they download the entire PDF and scroll through quickly to get an idea: print it, or close the window and look for other articles? That is the **third filter** and it's similar to being among the last few dream job applicants. What are the scientist's interests in the third round? Visuals, i.e. figures, tables, formulas, and chapter titles. Does your article make it through the third filter? Will the scientist use your article? We'll talk about figures another time, this chapter is about titles after all.

It can be slightly different when it comes to theses. Not every author wants people to read their thesis. We belive that there are people who wish for the opposite. Let's just work with the hypothesis that the author wants to write a good text, that mankind could find useful and worth reading. One that gets a decent grade.

3.6   Keywords – half the success

One of the best pieces of advice to write an article (scientific text in general) I heard is not entirely intuitive and obvious. **Write a list of keywords, that one should enter in the search engine to find your work as a relevant search result.**

Let your fantasy roam free. Think about how your work can be utilized and concepts connected to it. Write down all the keywords, this will be a couple of lines. Keyword can even be a phrase – typically two or three words.

Choose the important ones. This step needs some intuition and experience. I don't really know where you get those, but you can ask someone for help. By the time you write your thirtieth article, it'll be easy.

All the important keywords need to occur in the article title or in the title of chapters.

3.7   Title too general, title too specific

So what's up with the keywords? Titles are essentially pointers, ponting you in a certain direction: "The thing you're looking for is here!" For someone to appreciate your text, they need to not get lost in it. They need to know that the text offers answers to some of their questions. Titles can tell them this is the article they're looking for or to not waste time.

If you're moving and you write "stuff" on all of your boxes, you have truthful and formally correct labels, but you didn't really need to write anything. If they're too general, they're useless.

Chapter titles that contain one or two words are usually the primary suspect of a title too general – except for the conclusion, where chapter titles are canonical and I would avoid experimenting with them. If there are more single word chapter titles than the two mentioned, they're probably wrong.

Chapter title that could be used in a different thesis of the same branch e.g. "System implementation", "Image processing basics", "User interface principles" is suspicious of being too general. Something like "Fly movement monitoring system implementation", "Algorithms to detect objects and track thier trajectories" or "Principles of simple web systems user interfaces" is much better.

Chapter title that could be used on a completely different faculty is essentially always wrong – way too general. Title such as "Theory" could be used at a university of agriculture, IT, law, university of milk and cheese. It is wrong. Title such as "A study of existing solutions" is wrong. "Exploration of available technologies" is wrong.

I have never seen a title that is way too specific, and I don't believe anything like that even exists. It can be wrong – not describe, what the chapter actually contains. And if it does, it's never too specific.

I don't want to see five lines long titles. Vast majority of good specific titles will be on a single line and they'll contain roughly five words. Every now and then a title won't fit in a single line and there will be a good reason for it. Good titles – specific enough and not too long – are not easy to put together, and it requires thinking. Much like anything that should be worth something.

3.8 Abbreviations in titles

There is no place for abbreviations in titles, unless they're known worlwide (ČR, AIDS, IT).

You can explain a term in chapter one and say that it will further be referred to with an abbreviation. This abbreviation can then be used in the text of chapter without any further explanation. However, you can't use the abbreviation in the title of chapter two, because a scientist reads chapter titles in third filter when they decide whether or not to even read chapter one. If the scientist gets the feeling that the text is weird, not clear and that they don't really know what it is about (in case of abbreviation in title), they close the article and don't open it again.

Literature references and references to objects in articles (figures, titles, . . .) do not belong in the title and I've never seen a case where they would be needed (and I've seen quite a few cases, where they occured).

3.9 General advice from experienced supervisors

This subchapter contains a selection of hints from other experienced supervisors, whose students have successfully presented hundreds of academic works. They even took the time to write comprehensive articles containing their recommendations and posted them online. To read the full texts, you can visit their web pages, the links can be found in references [**?**], [**?**], [**?**], [**?**].

3.9.1 General advice from associate professor Beran

This subsection was taken from associate professor Beran's web pages [**?**], [**?**].

How to write/How not to write

- use chapter numbers up to second level, leave titles of a lower level without a number and don't include them in the table of contents, the final thesis and it's content is much less cluttered

- Logical structure – each object – whole thesis, each chapter, each subchapter has: introduction, treatise and conclusion:

    - introduction – tells the readed what it is about, what are the contents, what can we find out, what the problem is and introduces the context
    - treatise – deals with context, problem, details of a problem, solution, steps and the results
    - conclusion – summary of the task, achieved results and their meaning, concludes the work

- it's a scientific text, leave your emotions and opinions out of it

- don't use "WE" did, wanted, etc.

    - use passive voice, "tests were carried out" rather than "we carried out tests" – especially in the theoretical part, where thoughs taken from elsewere occur,
    - if you want to emphasize Your contribution, Your work, Your idea, etc. use "I" – designed a solution, experients, realization

- because WE (me-you, you-reader, you-world) didn't do anything, YOU did
- (ignore the fact that you're using my idea, that is expected, it is your thesis on my topic)

- when you use figures/ideas/tables from elsewere, **cite the source**

- each **title** (of a chapter or subchapter) should be followed by a paragraph of text that informs the reader about what they can find in the following section

- don't underestimate introduction and conclusion

3.9.2   General advice from professor Černocký

This subchapter was taken from professor Černocký's web pages [**?**].

1. Read a handful of good theses and try to understand what makes a good thesis. Your supervisor will gladly give you some examples.

2. **Czech/Slovak or English?**

   - If your english level is decent and there's a chance that someone else outside of BUT FIT will read your thesis (part of international project, work for an international company, SW description that you want to submit to GooglePlay etc.), I highly recommend you write everything in english. You can tell yourself that you'll translate it later, but there isn't really time for that. On the bright side, you don't have to worry about diacritics.
   - If you work on a thesis of a local significance and know, that your english isn't that great, I suggest you save yourself, your supervisor and the reviewer the trouble and write in czech or slovak. More information as well as common mistakes students make can be found in appendix B

3. Don't worry about the number of pages! Don't write about nonsense, don't copy needless things from wikipedia – this only makes your supervisor and the reviewer angry. If you follow the advised structure and write about what you have actually accomplished, the final product will be worth it.

4. Take your time – you don't have to write the text of your thesis in it's final state, but keep some sort of README file to write down what you're working on, current results, what you read, what was it about, etc. I highly recommend you avoid the "work first, write later" approach – in six months time, you won't know what you did and you'll have a hard time remembering or worse, you'll have to relive your past. Writing your thesis as you work helps you keep everything organized and structured.

5. Use a spell checker. Save your supervisor and the reviewer the trouble of correcting stupid errors (typos, etc). MS Word has a pretty good spell checker, linux has a decent ispall/aspell/hunspell utility (called from popular text editors such as Emacs). Some spell checkers are useless, e.g. the one in PSPad ignores a lot of errors.

6. Give your supervisor a draft of your thesis – in advance! – one chapter at a time is probably the best approach or at least two weeks before you submit the final version (tables with results/conclusions don't have to be finished). Your supervisor will cross

most of it out, you'll get mad at them, how dare they ruin your (nothing short of beautiful!) work. Once you cool off, you realize they're right, fix everything and the next version will be much better. If you instead opt to submitting a version that only you've read, it will severely affect the thesis review.

3.9.3 General advice from professor Zemčík

This section of text is taken from a document on professor Zemčík's web page [**?**]. Generally known and respected principles are written in normal font, whereas text in italics contains "extra" personal recommendations from the dean.

Table of contents should not be longer than a single page and an entry should not be longer than one "line". The thesis should be split in subchapters equally (except for introduction and conclusion). This principle even has a higher priority than the "Universal Decimal Classification" of a thesis[5]. When it comes to font, do not "waste" font styles and typefaces. Less is most definitely more in this case. Other than basic font, headings, captions of images, tables and equations, you should use as little fonts as possible, e.g. use italics or bold font style only to highlight text (preferably not both at once) and an appropriate "monospace" font for snippets of source texts. It is important to comply with the pre-written thesis formal template, that can be found on the faculty web pages.

As for introduction and conclusion, I highly recommend you don't split the in subchapters and keep them as concise blocks of text. The text above also suggests that it is appropriate to only use first level subchapters. If you ever feel the need to use more headings than an average of one per page (take a moment to think about if it is really necessary), you can use a "secondary" heading without a number that won't be included in the table of contents. When you're coming up with headings, consider the possibility that they might not be a good enough guide for the reader and if that's the case, you should probably change the headings. It's also a good idea to include a short text after each heading (i.e. chapter heading should be followed by "2. Work status" and rather than following up with "4.1 Initial status"), include a short text and then continue. Another thing you should try to avoid is ending text with an image or equation. It is advised to conclude a chapter with a brief summary.

Equations, images, charts, headings and others are significant typographical elements of a thesis. Their format, however, is for the most part "strictly" set by the template, which means that you don't have to spend too much time with them. Nevertheless, a few things should be said about their integration in the thesis. First of all, make sure that these "graphical" elements are well separated from the text and that the result "looks good". Excuses like "TeX did that" or "Word did that" will be short-lived. As for images, make sure you use uniform style of inclusion in text and that the images are either aligned to center or (if they are not as wide as the page) aligned to the inner side of a page (with respect to the future binding, or strictly right side in case of a single-sided print).

Note: If you aren't happy with what this chapter says about table of contents and you want to "guide" the reader better, make an index – it is a different typographical element and unlike the table of contents, you can use as many keywords as you want. It is, however, very time consuming and for that reason it's not something I'd recommend.

The primary language of a thesis is either czech, or english. Thesis cannot be in slovak "officially", but a thesis written in slovak with czech title and type is tolerated. Please, don't take this as some czech chauvinism, it's a matter of laws and academic field accreditation.

---

[5]Decimal classification according to ČSN ISO 7144 and ČSN 01 6910 - for extracts, see `http://web.ftvs.cuni.cz/hendl/metodologie/doporuceniupravydizprace.pdf`

When writing a thesis, use exclusively standard language and avoid colloquialism, or slang (technical slang included).

I highly recommend you pay extra attention to the following things, as they, unfortunately, are a common source of errors:

- *Try to avoid using first person singular ("I/me") as much as possible. The use of first person plural, despite the fact that it's commonly used in belles-lettres, needs to be eliminated completely. There are some exceptions though:*

  1. *You can use first person singular in introduction and conclusion of the thesis as means of stating your "personal opinion" (e.g. "Usually this method is used… , but I chose a different approach…"), it can also be used in the assessment of the current state, but never in summary of current state.*

  2. *First person plural can be used in case you're highlighting a part of the thesis, that you did not do on your own, but in a group. Considering the fact that a thesis should be primarily your work, you have to use it sparingly and it needs to be obvious, that at least 90 % of the thesis is your own work (e.g. "I wrote the program myself, but I asked my peers to help me with testing and together we conducted experiment…"). Avoid questions like "So you didn't work alone?"*

  3. *An exception to both rules stated above are mathematical texts, where first person plural is used often (e.g. "We have a cube of side…"), here's where you can use first person without any limitations.*

  4. *Another possible exception are rhetorical questions, if you ever use them in your thesis. I recommend using first person singular or first person plural at most ten times in the thesis (except for case 3, don't limit yourself there), there is no floor, but using it once or twice is fine.*

- *English expressions shouldn't really be used in a thesis. Considering the fact that our academic field is full of them, my recommendation would be to state both versions of a technical term (czech and english) if it is a first appearance and put the version that you will no longer use in brackets with a comment if you want (e.g. "… octree (oktálový strom, only the english version will be used going forward, because even experts in field are accustomed to it)…")*

- *Abbreviations should be explain when they first appear in the text. Alternatively (better, but more time consuming method), you can create a list of abbreviations, where all the abbreviations are explained in detail.*

- *Past/present/future tense should should be utilized as follows, generally speaking a thesis describes facts (use present tense) combined with a description of your work (that already happened, therefore use past tense). When it comes to future plans for the thesis, you obviously use future tense. However, more than anything else, use your "sense" and adjust the language, so that the thesis is easy to read.*

Chapter 4

Individual thesis chapters

In this chapter, We're going to talk about the meaning and recommended content of individual chapters a thesis. We have also included the recommendations from experienced supervisors.

Structure of a thesis changes depending on it's aim, progress made over time and achieved results. It's likely that you won't put all the things mentioned in this chapter in your technical report or that you will add an extra chapter that is not mentioned here. It's never a bad idea to plan the structure of your text in advance and consult your supervisor about it.

Individual chapters should be in a logical order. **Use references.** *In function XYZ, we implement mathematical formula, that we have derived in section 3.2, equation 7.* If you reference things further in the work, describe what you're referencing in roughly two sentences. The reader should not get lost in your work, don't make them flip pages. **Each chapter should (within reason) make sense on it's own.** If some important terms, abbreviations or thoughts are integral to the whole thesis and appear time and time again, always explain them (as soon as you use them in a chapter). If the reader opens your work in the middle (for example chapter 4), they should not immediately drown in abbreviations and technical terms.[?]

Unless specified otherwise, the rest of this chapter is taken from professor Zemčík's personal web pages [?], blogs of professor Herout [?] and assistant professor Szöke [?], and web pages of professor Černocký [?] and associate professor Beran [?].

4.1   Table of contents

Long text – such as a thesis – comes with automatically created table of contents. It MUST fit a single page in a thesis. I can see how it could be difficult to fit table of contents in a single page when it comes to a book seven hundred pages long, but not in a thesis.

In a thesis, table of contant should only have first and second level headings, third level headings are not present as they are a bit too specific. Fourth level – although it wouldn't be included in table of contents and only used in some chapters – is just wrong in itself.

If the chapter headings are good, just by looking at the table of contents (no further information, without reading the abstract) any reader that somewhat understands the field must be able to recognize what the thesis is about. They can gauge the goal of the thesis. They know what modules the solution consists of and what the purpose of each module is. They can tell which and how many experiments were carried out during the research.

They can also tell who the target reader of the thesis is – who and what is it useful for. If they can't tell just by looking at the table of contents, the chapter headings are probably wrong and it's up to the author to fix that, or have a poorly written thesis. There is no third option.

4.2 Introduction

The first chapter is titled Introduction. It's purpose is to provide broader context of a problem and define structure of the thesis in the form of an eptiome.

The length of an introduction should be roughly 1–2 pages of text. It is expected that the introduction is readable even by someone just "passing by", who is literate, lives in our time, on our earth and that's about it. They don't have to be an expert in the field. They should still be able to understand the introduction. It should be written in a way that makes it work as a standalone figure of literature and if anyone reads the it, they should understand what the thesis is about. It's also common that people only read the introduction.

4.2.1 Advice on thesis introduction from professor Zemčík

It is not appropriate to split introduction in subchaptes or include references in it – it should be a figure of literature that is readable and "feasible", and it should be easy to read. When it comes to the inner structure of an introduction, it should be as follows:

- Roughly 5 lines long general introduction for a topic, based on well known words without any scientific terminology if possible (example given "computer, video" yes, "tree structure" no),

- short text explaining why the thesis is important in the given field, the importance for world and for us, what the relationship with studied field is and other important facts,

- short text about the past advances in the field, about it's current state, and even about things to look forward to in the future,

- you should write about "why I am interested in this" too, what lead you to choosing this topic and do you find interesting about it – truthfully and honestly if possible,

- it's also a pretty good idea to write about the goals of the thesis (in your own words, not word for word from the specification),

- and finally, it is important to describe the structure of your thesis to make sure the reader knows where to look for things (example given "the following chapter contains..., described in chapter "xxx", in chapter 3 ..." etc.) — don't forget that the introduction should be a manual to your thesis.

4.2.2 Advice on thesis introduction from professor Herout

It is not supposed to be "an introduction to the problem", but "an introduction to a small book" (technical report). After reading it, the reader should

1. have a pretty good idea what the book is about,

2. look forward to reading it.

4.2.3   Advice on thesis introduction from professor Černocký

Introduction is basically an "extension of specification". It contains:

- Why was it researched – what is Your motivation?

- Project background – is it a part of a research project? describe. Any industrial cooperation? describe.

- If a thesis is a collective effort, be clear about who is responsible for what

- What are the specific things you would like to share ("claims").

- What can the reader read about and where. `\subsection{Table of contents}` etc.

4.2.4   Advice on thesis introduction from associate professor Beran

Introduction should be brief (1 page). It should contain:

- Introduction to the problem (we're in the IT field, for example: image processing and not chip manufacturing),

- goal of the thesis – one clear goal of a thesis and steps leading to it (there is only one goal, but there are many steps to reaching it (e.g. function library, creating a dataset))

- a brief summary of the entire thesis (an overview of existing solutions, introduce a draft of my solution based on the existing ones, testing, evaluation, ...).

4.3   Summary of the current state

This should be about 40–50 % of the length of your thesis. The purpose of the part is to familiarize the reader with the current state of technological field that is the subject of your thesis and introduce the apparatus used in thesis (mathematical, electronic, IT etc.) to them. It's not expected that the summary of current state will contain everything directly related to the thesis, but it should contain all the necessary information to make sure that a reader who is at least somewhat familiar with the field of your thesis can understands it. This part should be split in chapters, especially if it is a "multifield topic". This part should also "heavily" reference literature. The length depends on the type of your thesis, if it's theoretical this part will be much longer than in a practice-oriented thesis.

4.3.1   Advice on summary of the current state from professor Zemčík

It is appropriate to state at the beginning of this part what it contains and why, and that it is not "encyclopedic dictionary" of said field – to make sure that reader doesn't get the impression that "something is missing". You should write the text in your own words, but you can essentially copy the cited literature. If you have to take a whole section of text longer than one sentence, it's necessary to distinguish it properly from the rest and cite the source. A thesis should only have as many of these as necessary and the maximum length of such text is (roughly) half a page.

It is best that you don't express your opinions about the technical content in this part – the current state needs to be described, but not evaluated. Essentially, if someone wrote

something in literature, you can use it in this section. The reason for this is that if an expert in field reads the thesis, they should have the option to skip this part (they're an expert, I'm sure they're familiar with it) and not lose out on anything important in the thesis itself. It is advised to write this part as you make progress, while the literature is still in your brain – so you don't have to read it again.

4.3.2 Advice on summary of the current state from professor Herout

Chapter describing what needed to be studied should take about 45 % of the thesis.

I intentionally avoid the word "theory" that would fit this part of a thesis well. It's because the word theory has this magical property to trigger the urge to write purposeless text on various topics more or less relevant to the topic of a thesis, but also topics that are very distant.

You should ask yourself "Is this information necessary to understand what I designed and implemented?" whenever you're about to write a paragraph. If the answer is no, don't bother.

This part of the thesis text can often be comprised of two chapters. If you're developing a web-based accounting system, one chapter should be about accounting and the other one about safe web systems. Here's a typical example: lots and lots of IT solutions need to be studied as a field of work (where can the system help) and as a tool (process of developing such systems).

4.3.3 Advice on summary of the current state from assistant professor Szöke

This chapter (there can be more of them) is where you show that you understand the problem. You've studied the usual solutions to your problem, state why your solution is different from the others and why is it better. Theory is a great source of entries for the bibliography. Your grandmother does not need to fully understand this chapter, but it's a good idea to leave the mathematics for a second and summarize what your calculations and derivatives actually mean. It is especially useful should the reader lose themselves in the text, this will help them find their way back.

4.3.4 Advice on summary of the current state from professor Černocký

Theoretical part should be about 10 pages long, perhaps a bit longer when it comes to more theoretical theses. You should only write about the theory necessary for your thesis.

- Only describe what you actually need, we're not interested in contents of a script, book, wikipedia ...

- If you use formulas, you need to provide an explanation for every single symbol and it needs to be clear what the purpose of each formula in your thesis is. Don't use formulas "just to illustrate" or "to make it look more scientific" or "just to try it in LaTeX...".

- If a theoretical part is too complicated (for example HMM & co.), don't write about all of it, write about the basics instead and cite a good source to "pinpoint" what is absolutely necessary for your thesis.

4.3.5 Advice on summary of the current state from associate professor Beran

Theoretical part should contain:

- existing solutions from the perspective of your specification,

- what already exists in the field of my thesis, what other solutions of my specification exist,

- what existing tools and procedures could be used as a part of the solution,

- any and all theory should be **emphasized** (state why it is important that the reader is familizarized with it and how it relates to the solution your thesis offers).

4.4 Data

Data is the key feature of any project that deals with recognition and this chapter should not be missing. Recommended length is single digit of pages. Describe:

- where did you get the data (producer, catalogue number, etc.),

- technical description – e.g. Fs, bit width, number of speakers, audio length, etc.,

- dividing data in subsets – training, development, testing, evaluation – who did what (it's best to use existing subsets).

This chapter will probably not be a part of your thesis, unless you work on for example sound processing for real-time play.

This chapter will be a part of theses, that work with data sets, that are processed daily on different workplaces and allow for comparison of results, or theses that were given data sets for testing purposes.

4.5 Assessment of the current state and work plan (draft)

The main goal of this part is to write an evaluation of the current state and draft of your innovative solution.

4.5.1 Advice on assessment of the current state and work plan from professor Zemčík

It is advised to include this part of thesis. It's length should be "as long as you need it to be". The purpose of this part is to set a goal of your thesis based on the assessment of the current state and create your own "detailed specification", alternatively set the expected parameters of a solution, but not the the actual solution. This part should contain:

- Critical assessment of the current state (what is correct, what is wrong, what is not researched at all and possibly cost paramateres, availability of solution, necessary computing performance etc.),

- draft, what should be researched and solved based on the knowledge of the current state and personal preferences, specification, requirements etc.,

- given options even a specification of thesis, as in "what should it do", "what should the parameters be", "what tools will be used", "how is it going to be evaluated", "how do you know that you were successful".

It is advised to write a truthful deliberation in this part, especially when it comes to "draft" so that the rest of the thesis is credible and to convince the reader that all necessary steps were taken, after reading this chapter and the rest of the text.

4.5.2 Advice on draft of a solution from professor Herout

This part should contain ideas that the thesis brings:

- I decided to.

- I devised.

- I laid out.

- I calculated.

- I derived.

- I simplified.

- I improved.

- I designed.

- I found out.

- I researched.

Sometimes it's difficult to separate new ideas and implementation. Programmer confuses "designed" with "programmed". It's also easy to confuse "I improved" with "The results are". But separating these chapters is correct.

Theses in many non-IT fields are structured as research theses called "scientific methods". In our environemnt (lets assume engineer's decree IT study), theses don't follow this structure. Our theses (not entirely wrong in my opinion) are more similar to project documentation. It still is the correct way to separate hypothesis from a draft, how to validate them, and from their actual validation or assessment.

4.5.3 Advice on draft and description of your algorithm from associate professor Černocký

If the purpose of your thesis is "science", this part will probably be the longest one and it's advised to split it into multiple chapters – for example \chapter{Basis} \chapter{Innovation, ...} \chapter{Results and discussion}. On the other hand, if the purpose of your thesis is to try something existing/new, this chapter can be very brief. The length of this chapter should be roughly 10 pages and contain:

- what specifically did you do with the theory described above – block scheme, setting constants, technical simplification of complicated theory etc.,

- draft – can be a simple block scheme or a full object draft, but it should be clear that your SW has structure.

- choice of OS, programming language and libraries. The point of thesis is not to write everything on your own, you can use any and all free and commercial programs, libraries, modules, etc – essentially anything – it's a standard engineer's decree thesis – the goal is to **make it**, not to **write everything alone**. You do, however, need to describe everything in detail and cite sources, not plagiarize work of others!. Libraries have a good, accurate specification – where from, which version, if it needed to be paid for, how much did it cost.

4.5.4 Advice on draft of a solution from associate professor Beran

What it comes drafting a solution, it depends on the specification – the following points are not general. The length of this part should be about a third of the pages.

- Write from the perspective of a well paid expert on thoughts, that drafts a solution to problem, innovative solution, solution full of interesting thoughts.

- The "draft" part can be perceived as a procedure/tutorial to a solution, this draft is then passed to a team of programmers and testers, that realise and test the draft.

- If possible, it should be a general draft, without considering implementation on iOS or Android, Linux or Windows, MySQL or Postgres, HTML5 or Flash.

- Detailed specification analysis, detailed specification and formulation of goal and it's parts.

- Description of solution application, situations and problems that the project solves.

- Work procedure or steps leading to the goal, split the whole project into parts.

- Draft of the entire solution as well as it's parts, including references to theoretical part.

- Analysis of the results over time (measuring, observing, testing).

- Draft development and updates.

- If you update the draft based on the results of tests – include references to tests and their results.

4.6 User interface

This chapter is only useful in some theses and it should only be a handful of pages long. In some theses, it just doesn't fit at all. If it's necessary, it should be included (even prior to a draft) and contain: [**?**]:

- UI concept – what was the inspiration (existing programs, classic mechanical device...) – write about it,

- mockup – if you made any hand drawings, don't hesitate to include them!

- how did you choose the final version,

- if a UI went through more development, for example you weren't quite happy with the first version and made changes based on user reviews, write about it!

4.7 Implementation

This part of a thesis should be about the work itself. It should contain information about "what you actually did" and it should be about $40\,\%$ of the entire thesis. It should be clear what the basis of the thesis is, how was it made, what tools were used and what results were achieved.

4.7.1 Advice on describing your own work from professor Zemčík

When writing this part of thesis, it is necessary to avoid technical details that could distract the reader or even worse, bore them. Important things are concept of thesis, outlines of solution and what lead you to that solution. Another important aspect of this part is that it must describe how the solution is used, while not being a manual. Any and all technical details, that are not vital to understanding the thesis (and disrupt the "flow of the text") belong in the appendixes part and not the text itself. This mostly applies to long snippets of source code, instructions, tables of results etc. If you include snippets of your code in this part, keep them short and well distinguished from the text. A typical issue with this part is that it just isn't "feasible" for the reader due to a high number and depth of the solved details and the attempts to reflect how tough some thing were to deal with, in the text (usually in a way that emphasizes how much effort the author had to put in to finally overcome an obstacle), but the reader just doesn't care. On the other hand, images, photos or even screenshots are welcome. This part can be split in multiple chapters or it can exist as a whole. It is advised to split this part in multiple chapters if the realisation consists of vastly different parts (e.g. server programmed in C++ and a client in HTML etc.). The outline heavily depends on the individual in this case, although the basics can still be identified and should be in an order specified below.

Typical outline:

- What is the basic concept of the work,

- How does it work as a whole (and what is it good for), description of the functionality of individual parts of the solution (there's no need to emphasize everything equally – some things are more important than others, the "routine" can be reduced to a minimum),

- How do you use it including a good example, "case study" approach, "screenshots", procedures (instructions are not desired).

4.7.2 Advice on implementation from assistant professor Szöke

This chapter (there can be more of them) is where you describe your problem from implementation perspective. Which development environment have you chosen, which libraries, class design, communication design, protocol etc. Don't bother with the details. Instead of explaining how a button is implemented, explain how you have implemented artificial

intelligence, communication or an interesting function, that's what the reader is interested in. **There has to be a clear reason for every single decision you've made.** Again, your grandmother does not need to fully understand this chapter, but it's good to have multiple levels. Someone who is not completely lost in IT should understand what and why you have implemented, experienced programmer should understand even the details (how exactly have you implemented it).

4.7.3   Advice on implementation from professor Černocký

If the goal of your thesis is to create a "production" SW, describe it here. This part should contain:

- implementation comments – for example list of classes and what they represent, you don't have to describe minor things in detail (command line parameters), focus on the key functionality. Don't include full source codes here, those belong to the CD appendix. If a snippet of your code is vital here, you should include it and explain it's importance,

- if your program is supposed to communicate with the outside world in real time, write about the time sync, conflicts and how you deal with them. It's not expected that you make a 100 % "foolproof" SW, but you should know about the commonly occurring issues,

- if you implemented something in Matlab for example, write about it,

- what were the results – preferably comparison with existing results from someone else,

- if the goal was to build on something pre-existing and explore new options, here's the place to write about it in detail.

4.7.4   Advice on implementation from associate professor Beran

This chapter should be separated from testing, especially if the nature of the thesis is implementation heavy. The recommended length is about a third of the thesis.

Write from the perspective of a poorly paid programmer, that got specification of a prototype (your draft) and they're supposed to implement it. The chapter should contain:

- target platform and technology specifications,

- tools used for implementation of the solution prototype.

4.8   Testing

This text should provide a clear picture of how the functionality of the software was verified. Through mathematical or experimental means, or a study conducted on users etc. What were the results of verification? The length should be roughly 10 pages.

This part can have a single or multiple chapters. It's advised to split this part into multiple chapters if an extensive testing and evaluation was conducted. The outline depends heavily on the individual, but the basics can still be identified [**?**]:

- Methods and results of verification, that can contain mathematical proofs, testing procedures, testing procedudes involving humans,

- interpretation of results and possible application in practice (TODO things included).

4.8.1 Advice on testing from professor Černocký

This chapter chapter can be wildly different – only use things that are useful [**?**].

- Offline data testing – same or better results as the published ones? If not, why? Worse results don't necessarily mean that the thesis is wrong (access to less data, worse algorithm developed over the course of one semester, etc.), but you should know why.

- Offline data testing – are results from implementation in C/C++ and from the original Matlab algorithm the same?

- How demanding is it when it comes to HW? – CPU, memory, parallel behaviour, behaviour when using GP-GPU, etc.

4.8.2 Advice on experimentation from associate professor Beran

This chapter contains experiments and evaluation – if the nature of thesis is not implementation heavy, it can be included in the implementation chapter.

Write from the perspective of a poorly paid tester, that got prototype specification (your draft) and it's implementation and they have to test it. This chapter should contain:

- specifications on the platform used,

- experimental data used for the prototype – description of data, source, conditions under which the data were obtained, what are the data like,

- data annotation – annotation format, source and usage,

- description of measurements, description of experiments/tests and their conditions,

- measured data,

- results – discussion and interpretation of measured data.

4.8.3 User testing

Again, only relevant at times, but in some cases it is vital [**?**].

- Testing subject selection process (naive, experienced).

- "Testing protocol" – what did they actually test, what did you ask them – questions, evaluation, . . .

- Testing results – answers subject by subject and summary.

- Conclusions – Were the users satisfied? What were/weren't they satisfied with? Is it good? Is it bad? Can it be improved? Did you make any improvements during your thesis or is it a matter of future?

4.8.4 Advice on experimentation and testing from assistant professor Szöke

In this chapter, your results should be subjected to extensive testing. Not just you, let independent users test your solution. The most important thing is to collect all the feedback and formulate a relevant conclusion. For example improve GUI, make some sections of code faster or choose a completely different approach. If you have enough time, you can even try an addition iteration and work on the biggest shortcomings of your work.

4.9 Conclusion

The final chapter – Conclusion contains evaluation of achieved results with extra emphasis on student's contribution. A mandatory part of conclusion is also evaluation from the perspective of further advancement of the project. Student states ideas and suggestions based on experience with the project and also states how it relates to other finished projects (other bachelor's theses that year or projects at external workplaces).

The conclusion of a thesis should contain facts summarizing the thesis and give the reader (even without reading other parts of the thesis) information about what the subject of this thesis was and it's success. The conclusion should contain your personal opinions and impressions, and even better a summary of options going forward. Conclusion should be at most one page of text long. Don't include references to thesis text or literature in the conclusion.

Make sure there are no new breakthroughs, new numbers or a new chart in the conclusion.

4.9.1 Advice on conclusion from professor Zemčík

I highly recommend the following outline:

- Brief summary of the conclusion (for example "The goal of this thesis was ...".),

- Stating that the goal was achieved (preferably without self-criticism, save it for the reviewer),

- Summary of all satisfied requirements of the formal specification of your thesis, either directly as a "reaction to the specification" or "hidden reaction"; either way, state one sentence summarizing the answer to the question "How did you meet the requirement X of specification?",

- Healthy balance of qualitative and quantitative summary of the thesis, e.g 3–5 most notable information (numbers, data, etc.) about the thesis (recommended length is 5–10 lines),

- your observations ("I learned ..."),

- What are your future plans, preferably split in parts "I would like to continue my work and ..." and "My work could be expanded on ..." – different parts based on what you would like to try and what could be done, but you won't be the one to do it.

Please, keep in mind that that conclusion is a part of thesis, that people will read the most. If anyone reads your thesis in the future, they'll only read introduction, conclusion, or introduction and then conclusion, introduction, description of the work and conclusion, but it is very rare that people read the whole thesis. Each of the possibilities mentioned should be "feasible" for the reader.

4.9.2 Advice on conclusion from professor Herout

The functionality of a conclusion:

- Author looks back at their work: "The main accomplishments are. The important results are. I managed to."

- Author states ideas they did not have time to realise as a way to continue their work: "There are things that can still be done. If I knew then what I know now, I would."

- Author summarizes steps taken to satisfying the requirements of specification.

Two things.

- First: "Discussing ways to continue your work" sounds easy and safe. I would not take this brief statement lightly. Things like "It would be a good idea to increase the speed and precision" show, that the author does not really bother thinking about their work and that they don't really have any ideas. Other "ways to continue your work" can leave the reviewer thinking that you should have done it to meet the requirements of the specification, which means worse evaluation results.

- Second: Conclusion is the very last thing the reviewer reads just a few moments before they start writing their evaluation. Therefore the conclusion should put them in the right mood to write the best evaluation possible. It's a good idea to leave impression such as "I'm a good student, that met any and all requirements of the specification and did a good job." There's a thin line between that and "I'm a bad student, that can't think and program, so I have to scream that I'm the best out loud."

4.9.3 Advice on conclusion from professor Černocký

The conclusion contains:

- Summary of work – I did this and this, this didn't work, this worked, i didn't have time to finish this because . . .

- Future work

  - "short term" – things that you are certain you could do alone or with the help of 1–2 people within a couple weeks – months and that it would help.
  - "long term" – this is entirely a thing of your imagination.

Advice on conclusion from associate professor Beran

Conclusion should only be one page long and contain:

- what was the goal of the thesis,

- what steps did you take to get to your solution (see "brief content of the thesis" in introduction),

- what did you manage to create,

- evaluation of the solution based on results,

- other possible solutions, future possibilities,

- I do not recommend self evaluation and things like "I chose a specification, learned to program and achieved the goals of thesis".

4.10   Appendices

This section should contain descriptive parts of the thesis (e.g. user manual, snippets of source code, detailed schemes, descriptions of designed solutions etc). All appendices need to be numbered. If there are more appendices, there needs to be a list of them at the end of the thesis. [**?**]

There is no limit to number of appendices pages. However, it is necessary to keep it purposeful, concise and consider the significance of an appendix for the review of the thesis and for follow-up theses in the future. Needlessly large volume of appendices (i.e. not well justified) can negatively affect the assessment from at least environmental and economic perspective. [**?**]

Appendices are the right place to include instructions, detailed descriptions of designed protocols and formats, tables, most images and other elements that would disrupt the "flow of the text" of thesis. The length of appendices does not count towards the length of the entire thesis. The number of pages should not be too high – it is important that all the appendices in paper form serve their purpose.

4.10.1   Advice on appendices from associate professor Černocký

Appendix one – The cookbook   Instructions for those who would like to redo your work, less than 10 pages

- What needs to be downloaded, compiled, how to "hack the OS" to make everything run smoothly . . .

- directories, scripts, launch order, where to look for the results,

- etc.

Other appendices   Anything that would disrupt the flow of the text of thesis – a perfect example is a two pages long derivation of something, three pages long table of specifications, etc. . . .

The included data medium should contain:

- All the source codes – Matlab, C, LaTeX, etc.,

- All the parameters of a model – HMM, neural network, transformation, basically everything,

- Everything necessary to be able to launch your software – external libraries, modules, etc.,

- All the data – unless they're under some sort of license,

- Detailed results – table summarizing the results belongs in a thesis, but you can include multiple MB of automatically generated tables in the appendices

4.10.2   Advice on appendices from assistant professor Beran

Assistant professor Beran recommends you include:

- all used libraries, source codes and build instructions,

- Your application (including binaries), ie. solution executable directly from the CD,

- Video – a decent presentation of your software, you can even include a clip of a working final version of your software,

4.10.3   Poster

- Burn it to CD and print it too (PDF is probably your best bet),

- A2 paper size,

- print it in the library or have it printed by a commercial printing company,

- A healthy balance of the contents:

  - preferably eye-catching and clear format (what is it, what can it do, what is it good for, why is it good),
  - a bit of technical description (procedures and methods used).

- A poster should have:

  - student's first and last name
  - student's email
  - supervisor's first and last name
  - academic year

Chapter 5

Rules for bibliographic citations

These rules were taken from the faculty web pages [**?**].

5.1 Definitions

- **Bibliographic citation** is a summary of details on the work cited, or its part, allowing for its identification and search.

- **Reference to a bibliographic citation** is an in-text reference to a bibliographic citation in a different part of the work.

5.2 Using works of others

When using works of other authors, it is necessary to properly differentiate them from your own text. Otherwise you present the text as your own, which is an unacceptable violation, especially when it comes to theses and dissertations. There are two ways to incorporate works of other authors into one's own:

- **Quotation** – the text used matches the source document word for word. Short quotations are enclosed within double quotation marks, long ones in an indented free-standing paragraph. The quotes are also often typed in italics.

- **Paraphrasing the original text** – the author of the presented work puts the original text into his/her own words, while maintaining the meaning of the original statement. Paraphrases are typed in the same typeface as the rest of the text. In order to distinguish which part of the text is borrowed and which is your own, you must use other means (for example, by a reference to a citation at the end of the sentence or paragraph the author indicates that the sentence or paragraph was cited or paraphrased; when the paraphrase is more extensive, you can specify that at the beginning of a subchapter, for example: "This subchapter has been adopted from [1]".)

Quotations are suitable for definitions, parts of laws, regulations or rules, or to discuss the opinions of other authors. Quotations are therefore not that frequent in technical texts. If there are no valid reasons for a quotation, use a paraphrase. Also, think about the scope of the quotation. Too many and too long quotations indicate that the thesis is of poor quality (usually finished in the last minute, and the quotations only serve to reach the minimum length required).

5.3 Basic principles of citation

Bibliographic citations are provided to make it clear to the reader what works the author used as a basis for his/her own, to familiarise the reader with a wider context (for example, so that a reader less knowledgeable than the author's target audience could find more on what the author did not explain in his/her work) and also to comply with the terms of the Copyright Act (see Section 31). When citing, the following rules must be followed:

- **Cite all the sources you have used!** If you fail to cite any of the sources, you are presenting someone else's work as your own! Make notes of all the sources you have found since the very beginning of your work on the project. It is more difficult to look up all the sources once you are done.

- **Cite only the works that you actually used!** Readers who actually know the works cited (or look them up) will soon realise that even though you use them as reference, you do not really know them.

- **Only cite primary sources!** Only cite the works that you actually worked with `physically` (or on a screen). Otherwise you risk using a quotation with mistakes.

- **Cite precisely!** You will make it easier for the readers to find the original source avoid the suspicion that you intend to make the identification of the source difficult or impossible by providing an incomplete citation, so that the readers cannot check the scope of the quoted text.

When writing your text, pay special attention to these citation principles, as their violation means, for apparent reasons, far more serious consequences for the overall assessment than non-compliance with the formal requirements for bibliographic citations.

5.4 Citation standards

The standards for creating bibliographic citations and their reference in professional publications are governed by ČSN ISO 690:2022 (Information and documents – Rules of bibliographical references and citations of information sources) of 2022. The standards are available at the FIT Library. There is a citation generating tool based on these standards, available at http://www.citace.com/ (in Czech, but simple to use – write ISBN, DOI or name and search it). and you can find a brief summary of the standards with citation examples on the same website [**?**].

Although the standards allow to place citations in professional publications in different ways (at the end of the text, at the end of the individual chapters, directly in the text, in the footnote, partly in the text and partly in the footnote), it is required for theses and dissertations at FIT BUT that the list of works cited is provided at the end of the work.

The reference to the citation is in the form of a serial number, under which the citation is listed in the Works Cited section at the end of the work. The serial number is stated in the text in square brackets.

Example: *The SMTP protocol is specified in RFC 5321 [1].*

Citations are in alphabetical order. Ordering of names containing letters with diacritics can be changed using the `key` element, the value of which can be set to a last name without diacritics. If a record does not have the author element, the citation is ordered to the

beginning of the list, which is not ideal. In this case, we can change the ordering by setting the key element to a desirable value.

Example:

```
@Article{Cech:2020:Citace,
author            = "Čech, Jan",
key               = "Cech",
...
```

If you quote the same publication several times in a row, you can use the term "Ibid." instead of repeating the entire reference, supplemented by a page number.

Examples of citations are in appendix F.

5.5 Using electronic sources

Finding high-quality sources to use as a basis for your work is essential. The quality of a classic printed work may be recognised quite easily. When working with electronic sources, examine their credibility and quality thoroughly. Always verify who the author of the material is, and evaluate the depth of the information provided (a number of electronic sources presents merely superficial and incomplete information, and therefore the authors copy eachother's mistakes) critically. Even if you find high-quality electronic sources, do not rely solely on those, and try to find a printed work as well.

Whenever you cite electronic sources, always state when the information was found and used as the entire website may not be available several days later. When using BibTex, this can be achieved by adding the following key: (`cited = "yyyy-mm-dd"`). You can use the `@website` (entire domain) or `@webpage` (a single page within the domain) record types to cite electronic sources. If you cite a magazine article or a paper from a conference, do not cite them as electronic sources and use `@article` instead.

5.6 Evergreen: citing web vs. paper

Yes, everything is on the web these days. Thank god.

Web however does not have an archive. It changes every minute. If the references to literature are all URLs, this part of is somewhat alive, ungraspable, fluid – links refer to an environment, that changes every second. This part of thesis should ideally be rigid, constant – only references to paper version of literature.

It's easy to cite a web source, when it comes to a downloaded .pdf file of a magazine or conference article. Don't do it in your thesis.

Don't include this in the list of referrenced literature
BAY, H., ESS, A., TUYTELAARS, T. and GOOL, L. V.: Speeded-Up Robust Features (SURF) [online]. 2006, updated 2008 [cit. 2010-07-13]. Retrieved from: https://www.vision.ee.ethz.ch/en/publications/papers/articles/eth_biwi_00517.pdf

when you can use this instead
BAY, H., ESS, A., TUYTELAARS, T., GOOL, L. V.: "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346–359, 2008.

There's a chance that the web no longer works in a month, whereas a conference proceedings can be found retroactively until at least the end of this civilization.

5.7   Typesetting of citations

The template for BUT FIT theses and dissertations uses the BibTeX system to typeset citations.

The following text is taken from ČSN ISO 690 standard and condensed. Individual parts of a citation are separatedy by the dot symbol ".". Each item in the cited document should be stated as it appears in the original cited document. The usual order of items in a bibliographic citation is as follows:

1. **name(s) of the author(s)/creator, if they are available(s)**

   - Persons or institutions responsible for creation of the contents should be stated as authors.
   - First names and other parts of names should follow after the last name, e.g. GORDON, D.
   - If a cited source has multiple authors, list them (using the "and" separator), and if there are more than three authors, we do not list all of them and use "et al." (names followed by "and others") instead.

2. **title**

   - Title is typed in *italics*. A title too long can be cut short and missing words replaced with ... (three dots).

3. **type of medium**

   - A type of medium is a medium, where the cited material is available (e.g. [DVD]) as well as the form in which it is available (e.g. [Braille]). This information is stated in square brackets. [online] is used for online materials.

4. **edition**

   - Edition should be stated with symbols, e.g. "2nd ed., (re-edition)". If it is an updated version, it should be apparent from the citation, which version was used.

5. **place of publication, publisher, date of publication**

   - If the place of publication is not stated in the cited document, but is known, we state it in square brackets.
   - A publisher should be a person or organisation that is the most emphasized in the cited document.
   - Date of publication should always be stated. A year is sufficient for paper publications, however it is necessary to also state a month and a day for electronic sources.

6. **numbering within a unit, page or page range**

- Citation should idenfity the specific part of a cited document, that is used for citation.

7. **title and volume in series, if they are available**

8. **standard identifiers**

   - If a cited document has an international standard number (ISBN, ISSN, ...) or DOI, that serves as the unique identifier, it must be stated.

9. **dostupnost, přístup nebo umístění informací**

   - The location of source should always be stated for electronic sources. Stated as: "Available at:" followed by URI or URL.

10. **additional general information**

Chapter 6

Formal aspect of a thesis

The formal requirements for writing a bachelor's or diploma thesis are based on rector's directive no. 72/2017 [**?**] and FIT guideline no. 7/2018 [**?**]. Formal aspect of a thesis is an important part of reviewer's assessment and should be given some attention (it is necessary to familiarize yourself with the mentioned directives). Other instructions and recommendations are listed on faculty web pages [**?**], [**?**].

Required length of the thesis text without appendices is shown in table 6.1:

Table 6.1: Required thesis text length in standard pages

	Minimal length	Usual length	Length should not exceed
Bachelor's thesis	40	60–80	100
Semester project (SEP)	20	30–40	50
Dissertation	50	80–100	120

The length of typeset pages will be roughly 1/2 of the extend in standard pages. The term *standard page* applies to evaluating the volume of a thesis, not the number of printed sheets of paper. From a historical standpoint it's approximately the number of pages of handwriting, that used to be typed on a typewriter to pre-printed forms at the usual line length of 60 symbols and 30 lines per page. Considering the need for proofreading marks, the line spacing value used to be 2 (every second line). This data (number of symbols per line, number of lines and the spacing between them) are in no way related to the final printed product. Their only use is to assess the length. **Therefore, one standard page means $60 \cdot 30 = 1800$ symbols including whitespaces.** Images inserted in text are counted towards the length of the thesis by estimating the amount of text it replaces in the final document.

For a rough estimate of the number of standard pages when using the LaTeX system, you can use the sum of source file sizes of the thesis and divide it by about 2000 (usually we'd divide by 1800, but source files contain other symbols and commands that do not count towards the length of the thesis). For a more accurate estimate, you can extract the plain text from a PDF (e.g. use cut-and-paste or *Save as Text...*) and divide it's size by 1800. You can also use the Detex[1] application (on Linux available in it's distribution, for Windows, however, you need to install it separately[2]), that removes special symbols

---

[1] https://www.ctan.org/pkg/detex
[2] http://urchin.earth.li/~tomford/detex/

and commands from source text and then you can divide it's size by 1800. **The Detex application uses the `Makefile` in this template to count the number of standard pages in the core of the thesis too** – command `make normostrany`.

In Microsoft Word the approximate length of the thesis in standard pages can be computed using the *Word Count* function in *Tools* menu if you divide *Symbols (spaces included)* by 1800. Only the core of the thesis is counted towards it's length. Parts such as abstract, keywords, declaration, table of contents, references or appendices do not count towards the length of the thesis. It is necessary to select the core of the thesis first and then have the software count the number of symbols. Assess the length of images manually. The same procedure can be applied when using OpenOffice or LibreOffice.

Original text dealing with the assignment, where the solution is core of the thesis, must be at least a third of the entire thesis text. A mere compilation of available sources is unacceptable.

The subject of reviewer's assessment is primarily the text and final product. Needlessly large number of pages is the evidence of poorly processed topic and a burden for the reviewer. Theses, where the length of text report is equal to amount of work done can be longer and enriched with explanatory text. The explanation of the essence of solved problem and the procedures used to solve it does not have to increase at a linear rate with the amount of work. Well structured and comprehensive text report can be of relatively small length. Detailed descriptions of significant parts of project, that are more of a documentary (rather than explanation), can be included in appendices and referenced from the main text.

If the length of a text report is close to the minimal required length, reviewer will focus heavily on whether or not individual parts of the report for understanding the thesis are really necessary. The inclusion of foreign texts, that only vaguely relate to the topic of the thesis or ones that are questionable, in attempts to reach at least the minimal required length (for example not enough time just before the submission deadline), can lead to a much worse overall review of the thesis.

When you insert an image, choose the dimensions carefully so that they do not overlap the are for text printing area (i.e. text margins on all sides). Put large images on a separate page. Images or tables that are larger than A4 dimensions can be folded (so-called Engineering fold, similar to Z-fold but doesn't make issues with binding) and included in the main text (if it is crucial for the thesis), appendices or fold it and put to the pocket on the back side of the binding.

Images and tables use their own, independent numerical series. This means that references in the text must state if it is an image or a table as well as the number (for example "... *see table 2.7* ..."). It's rather natural to comply with this principle.

When it comes to references to pages, chapter and subchapter numbers, image and table numbers and many other examples, we use special tools provided by the desktop publishing program, that can generate the correct numbers even if the text moves due to changes made to the text itself or changes to typesetting parameters.

Equations referenced in the text will be provided with serial numbers on the right side of their respective line. These ordered numbers are surrounded by round brackets. Equations numbering can be in order within the whole text or within individual chapters.

If you are in doubt when printing a mathematical text, try to follow the LaTeX-defined printing system. If your thesis contains a large number of mathematical formulas, we highly encourage you to use LaTeX system.

There is no space between a number and letters that form a single word or a symbol – for example *25x*. In a paragraph, it is better to spell both out though – for example *ten times*.

Punctuation symbols such as a dot, comma, semicolon, colon, question mark and exclamation mark, and even closing brackets and quotation marks are adjacent to the preceeding word, without a space. The space comes after the punctuation symbol itself. This, however, does not apply to decimal point. Opening brackets and quotation marks are adjacent to the following word and the space in front of them is excluded – (like this) and "this".

We do not use the same symbol for connecting and separating dash, and a regular dash. There is a special (longer) character for a dash. In TeX(LaTeX) system, the connecting dash is a single "dash" symbol (e.g. "Brno-center"), when typing text for intervals or pairs, opponents and others we use two "dash" symbols (e.g. "price 23–25 crowns", "match England – Belgium"), to separate a section of a sentence, to separated an inserted sentence expressing unspoken thoughts and in other situation (see orthography) we use the longest type of dash represented by three "dash" symbols (e.g. "Another term — no matter how pointless it seems — will be defined informally in the following paragraph."). When typesetting a mathematical minus symbol, we need to use a different symbol again. In TeX system, the source text, the regular minus symbol is used (i.e. "dash" symbol). Typesetting in math mode, however, is done by surrounding the formula with dollar symbols to generate the correct output.

Rules for the use of abbreviations in different languages are listed in their respective orthography books (e.g. Pravidla českého pravopisu [**?**]). There are other reasons to always have one of these close.

6.1 Common errors

This chapter contains a selection of the most common errors as well as advice on how to avoid them, taken from professor Herout's blog [**?**] and from the list of common errors that assistant professor Szöke posted on their blog [**?**].

Minor things that notoriously ruin reading

- **Using hyphens instead of dashes**
  A dash is long and there should be whitespace surrounding it. Dash is very often used instead of a dot in sentences: "This book – published before the war – is amazing." It is also used for ranges: "page 23–26" or "success rate of 3–5 %." More examples can be found in the language handbook [**?**].

  Hyphens occur in our IT theses very rarely (at least they should). A good example would be phrases or joining subjects "Rh-factor", "real-time", "propane-butane".

- **Brackets surrounded by whitespaces**
  There always is a whitespace in front of the opening parenthesis or brace (when referencing literature). There is no space after the closing parenthesis or brace, if they're immediately followed by a dot, comma, exclamation mark or question mark. There are no whitespaces inside the brackets.

Here's a brief overview of common stylistic and language sins.

- Correct in english:

- "by using the OpenGL library"
  - "in the MVC model"
  - "all UI elements"
  - "from the JSON string"
  - "call it from C# code"

Incorrect in czech:

  - "s použitím OpenGL knihovny"
  - "v MVC modelu"
  - "všechny UI prvky"
  - "z JSON řetězce"
  - "volat ji z C# kódu"

Correct in czech:

  - "s použitím knihovny OpenGL"
  - "v modelu MVC"
  - "všechny prvky UI" – nebo ještě radši "všechny prvky uživatelského rozhraní"
  - "z řetězce ve formátu JSON"
  - "volat ji z kódu v jazyce C#"

- **Sentences without verbs**
  Every single sentence needs a verb. Sentences without verbs are sometimes used for artistic purposes in poetry. Over the course of the last two weeks, I have read countless sentences without verbs in them and it never quite worked, always lead to a negative outcome. Make sure every sentence of your thesis has a verb in it.

Figures (almost) without captions

Figure (similar with a table) consists of the image itself and it's caption (`\caption` in LaTeX). The caption of a figure or a table is there to make sure the final object works as a standalone object – reader checks the figure often even before reading the surrounding text and therefore it is expected that they can understand the figure even without reading the text.

I wouldn't worry about five to seven lines long figure captions. Two lines will do just fine at times. Sometimes – not very often – captions consisting of just three words are the best ones. If all the captions in a thesis consist of only three to four words, the reader will most likely be frustrated, because the figures make no sense whatsoever.

If a figure contains a colorful code (some objects are red, some are blue and others thick green), proper explanation of code must be a part of the caption. If a figure consits of parts (for example top right, top left and bottom), naming and explanation for each part belongs in the caption.

Whenever I tell my students this, they get anxious about the whole thing: "But that means I have to move entire sentences from text to the captions!" Yes, move them, there's nothing wrong with that. The basic explanation will be a part of the figure and that's how it should be. More detailed explanation, reasoning and interpretation should remain in the text itself. Twenty lines long caption for a figure or table is too long, but five lines long caption is a standard.

Grammatical person

Using **second-person** (addressing the reader with "You/you") is almost always wrong and it gets annoying very quickly.

Incorrect:

- "If you look at figure 5, you'll see ..."

- "If you work with library X, I'm sure you'll stumble upon ..."

- "If you want to switch to settings, select the respective option in menu."

Correct:

- "Figure 5 shows ..."

- "A commonly occuring thing when using library X is ..."

- "Settings can be accessed by selecting the respective option in menu."

Using **first-person plural** ("we") is not always wrong either and "classic" literature about writing technical text sometimes even recommends it as so called "author's plural":

- "We found out ...",

- "We focused on ...",

- "We designed a solution, that ...".

Less experienced writers often switch from author's plural to some weird and incorrect language mode, that has a common occurence in kindergarten. This is how teachers in kindergarten speak (nothing against them otherwise): "Alright kids, let's glue a bead right in the middle of the flower. Now we push out a bit of glue, yeees, and we push the bead into it with a finger, juuust like that."

It's not funny and cute, if diploma student uses this language to describe their life's work: "First we need to link library X. Then we create objects of selected classes and send them to a server one by one. When the server responds with an error code, we must reset the connection." Do not use this language mode if you wish for your thesis to be perceived as a work of someone who is not on the mental level of a child in kindergarten.

Using **first-person singular** ("I") is correct, when it comes to a matter of subjectivity:

- "I focused on ...",

- "I created ...",

- "I measured ...",

- "I addressed several respondents ..."

It is incorrect (and unfortunately quite common) to use first-person in the description of procedures and phenomena:

- "In the first step of the algorithm, I reset counters."

- "If the pointer points to null, I allocate new object."

- "It's clear from the graph that the cache size I used is too small."

Other errors

Don't cite stupid things: Information in introduction about the fact that internet is available even on Mount Everest can seem like a thing, that needs to be confirmed. Nevertheless, this pointlessly inflates Literature with a lot of irrelevant sources (not within the scope of your thesis) – that can be a bad thing for you. Just don't write anything like that in your thesis, or don't cite it at least.

Abbreviations (and notes) in footnotes should make sense on their own. If your text contains an abbreviation, specify what it stands for when you first use it: If an abbreviation occurs two chapters later, you need to specify what it stands for again when you first use it there, alternatively write a footnote addressing the abbreviation. It is important that the footnote contains the abbreviation too. If it's an abbreviation of a phrase, feel free to translate it to the desired language.
Incorrect:

1. large-vocabulary continuous speech recognition

Correct:

1. LVCSR – large-vocabulary continuous speech recognition

Don't leave text between a section and subsection out: There should be a text between the titles of chapters and subchapter. This calls for $1 - 2$ paragraphs, where you explain what the chapter is about and what can the reader learn here.

Hand drawn sketches and images: It's easier to quickly draw something, take a picture of it and continue writing, especially when it comes to a work draft. Usually the initial sketch of a scheme is wrong and re-doing it takes precious time. I'd be careful with "hand work" in the final text. Some reviewers could see it as a false indication, that you did not have enough time, and so you decided to save time by doing some quick hand drawing.

Introduction and conclustion: Typos, nonsensical sentences, errors, ungrammatical words, . . . Neither one of these belong in a thesis. An evasive typo can hopefully get lost deep in a thesis. Everyone reads the introduction and conclusion though. Making errors here is shameful. When you try to convince the committee members, that you did a big chunk of work, how is the fact that you can't read a single page of text going to affect the thesis review?

Cite figures: If you "borrow" an image from a source, don't cite it and it is later discovered, you receive an F. Citation referring to literature at the end of the figure description is (probably) OK. Nonetheless, many reviewers prefer stating the source explicitly in brackets: (taken from literature [xy]).

Don't overdo it with chapters: If a chapter has $1 - 2$ paragraphs in individual subchapters, try to think of a better way of doing it, like bullet points.

Notes in images: Don't describe complicated images with "A is on the left side, B is at the top, C is in the middle, D is under that and next to it is XY." This results in a half of a page worth of text. And if the figure is not on the same side, the reader has to flip the pages til they die. Add captions to figures and it becomes a standalone figure.

Details belong to appendices: Don't try to describe a complete class diagram, table diagram, object diagram, ... at any cost. Include one complete scheme and describe the basic components. Then you can focus on the core of the system (for example the three highlighted objects). Those are the key and they're the basis of your thesis. Description of other "supporting" objects (e.g. data loading and rendering) can be a part of an appendix.

You can simply reference the appendix by writing: *Detailed description of all classes and their methods can be found in Appendix 1.*

Avoid compound sentences that are too long: If your sentence is the length of a paragraph (6.5 lines), by the time the reader reads third line, they already forgot the beginning of the sentence. Don't be afraid to split the compound sentence into smaller parts. Fire short sentences at the reader rather than suffocating them with loads of words. Alternatively, you can present information using the `itemize` LaTeX environment.

Unbreakable spaces: A typographical error such as leaving "s" at the end of a line as early as the title of your thesis is extremely shameful. There is something called non-breaking space, in other words a space, that prevents breaking the end of a line. In LaTeX, this space is generated using a tilde ~. It is described in most LaTeX books.

Chapter 7

Thesis submission

Bachelor's and diploma theses are submitted in both printed and electronic version. The data medium is included only if you have approved delayed publication, otherwise the electronic version is submitted via BUT IS. It is only considered submitted when both versions of a thesis are submitted properly. This chapter is mostly taken from the official instructions that can be found on the web [**?**], [**?**].

Those that have decided to keep some of the parts of their thesis a secret need to submit an application at least a month prior to the submission of thesis. A paper stating that it is a print with delayed publication, length of the delay (at most 5 years), and the reason for the delayed publication will be freely embedded to the thesis. The thesis will then be published when the agreed upon date of delay has passed.

Before you submit your thesis, go through the checklist (appendix C) and make sure everything is okay. Then make sure that the thesis is in compliance with directives [**?**] and [**?**]. Important details that everyone seems to forget from time to time:

- **Title page:** Don't forget to set the correct year (of submission) and department according to the specification.

- **Specification:** Don't forget to download electronic version of the specification (template expects a file named `zadani.pdf`).

- **Declaration:** Don't forget to sign the declaration in both prints of the thesis.

- **Bibliographic citations:** Make sure you cite all used sources in text.

Print the thesis and have it bound. As soon as you open the bound thesis on a random page, you'll find an error. Honestly, don't worry about it, that's just how it is. Nothing is perfect, so "just ship it"[1] [**?**].

7.1 Submission of printed versions of thesis

Bachelor's thesis or dissertation is submitted in one print, with signed declaration of authorship. The submitted print must be bound in an permanent way. A file made out of dark (blue, grey) bond paper is recommended. If the vice-dean has approved your request for the delayed publication, there will be an envelope with a CD/DVD or other allowed media glued to the inner side of the file back so that it can be easily accessed.

Student also submits the following parts of thesis in electronic format:

---

[1] http://blog.igor.szoke.cz/2011/08/zkoumejte-bezte-za-bod-odkud-neni.html

- text report in PDF format (and in BUT IS),

- source form of the text report (including everything necessary for editing and printing again),

- complete documentation (installation instructions, user manual, schemes, etc.),

- source codes of programs (binary programs must be compilable from submitted source files),

- all programs in executable form capable of running in FIT Computer Center environment. If this cannot be achieved (machines in the Computer centre do not have the required software installed, or the required hardware is not available, or if the result of thesis is a new hardware), after discussing the situation with their supervisor, the student will present a functional product to the reviewer.

You can convert document from LaTeX to PDF using the `pdflatex` application or from `dvi` using `dvipdf` application, or even from PostScript using `ps2pdf` application. The size of a final document should be under 10 MB. If it is larger, something is wrong. Documents like this usually contain needlessly large images.

A complete electronic form of thesis must be included on a non-rewritable media CD-R, DVD-R, DVD+R in ISO9660 format (with RockRidge and/or Jolliet extension) or UDF, or SD (Secure Digital) memory card in FAT32 or exFAT format,and card is set to write-protected mode.

7.2   How to submit thesis in BUT IS

The full thesis text in PDF must be submitted to BUT IS – in the My Final Project module. You also need to fill in the abstract and keywords, both in Czech (Slovak) and English (be sure to replace non-breaking spaces when copying from LaTeX). The only reason to not make the thesis text public immediately is protection of intellectual property, which must be approved by the vice-dean. In this case, the student submits the full version of the work for review in writing and electronically. Thesis is considered submitted only when all the requirements are met.

The semester project for master's thesis is also submitted to the BUT IS. Bachelor's term project (term thesis) is only submitted if the supervisor requires it.

Chapter 8

Conclusion

This text summarized the formal requirements for a technical report of a bachelor's thesis or a dissertation. It described the usual procedures used when writing a text of technical nature and offered additional information and independent useful hints and tips for creation of a technical report of a dissertation. It was also explained that a bachelor's thesis also a dissertation and needs to be approached as such.

It is necessary to point out that dissertation is a unique individual work, that is developed under the supervision of an experienced expert. Regardless of what this template says, you're only obliged to comply with the official guidelines stated on the faculty web pages. You always need to consider which things in the text above are relevant for a specific dissertation and which are not. Most importantly, you should listen to your supervisor, who understands the given problem the most and is therefore able to provide the best advice that you can get.

Despite the effort, it is not possible to include all the elements needed for developing a thesis in this template and guarantee that once the text, images, literature and others are added, that everything will be alright for every single dissertation. A longer text than expected will break to two lines, en entry in list of references that the style was not tested with, and in other cases the result can be hardly satisfying. It could require a modification of the template to account for an error that occurs once in hundred projects. The final PDF and consequently the printed version needs to be thoroughly checked, don't let thoughs like "this was generated by the template, therefore it must be correct" cloud your judgement. If you find errors in the template or you have suggestions on how to improve it, contact us via email at `sablona@fit.vutbr.cz` and help us improve it. Any and all comments and suggestions are welcome.

Your supervisor can help you significantly when it comes to correcting errors. However, do not expect them to read through your work the night before submission deadline. For that reason, it is necessary to have everything ready in advance and consult your supervisor as you write your dissertation. Supervisor's critical viewpoint can allow for a better result and the extra effort will have a positive effect on their evaluation of the work.

Lastly, on behalf of all the authors, I would like to wish everyone currently in development of their own dissertation and those who are getting ready to start developing it a successful completion and presentation of their work.

Bibliography

[1] BELAIRE, N. et al. *Automatic LLM Red Teaming.* 2025. Available at: https://arxiv.org/abs/2508.04451.

[2] ESMAILZADEH, Y. *Potential Risks of ChatGPT: Implications for Counterterrorism and International Security.* 2023. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4461195.

[3] OPENAI. *Red Teaming Network.* 2023. Available at: https://openai.com/blog/red-teaming-network.

[4] PARLIAMENT, E. and COUNCIL. *Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act).* 2024. Available at: https://eur-lex.europa.eu/eli/reg/2024/1689/oj.

[5] PURPURA, S.; LI, A. et al. *Building Safe GenAI Applications: An Overview of Red Teaming for LLMs.* 2025. Available at: https://arxiv.org/abs/2503.01742.

[6] SCHOEPF, M. et al. *MAD-MAX: Modular Adversarial Red Teaming of LLMs.* 2025. Available at: https://arxiv.org/abs/2503.06253.

[7] WEI, A.; HAGHTALAB, N. and STEINHARDT, J. *Jailbroken: How Does LLM Safety Training Fail?* 2023. Available at: https://arxiv.org/abs/2307.02483.

[8] ZOU, A.; WANG, Z.; KOLTER, Z. and FREDRIKSON, M. *Universal and Transferable Adversarial Attacks on Aligned Language Models.* 2023. Available at: https://llm-attacks.org.

Appendix A

How to use this template

This chapter describes individual parts of the template, followed by a brief instructions on how to use it. If you have any questions, comments etc., feel free to email them to `sablona@fit.vutbr.cz`.

A.1 Template parts description

Once you extract the template, you will find the following files and directories:

bib-styles Literature styles (see below).

obrazky-figures Directory for your images. Currently contains `placeholder.pdf` (a.k.a TODO image – see below) and image keep-calm.png to demonstrate inserting raster images (you don't submit these images with your thesis). It is advised to use shorter directory name, so that it is only in your chosen language.

template-fig Template images (BUT logo).

fitthesis.cls Template (design definition).

Makefile Makefile used to compile the project, count standard pages etc. (see below).

projekt-01-kapitoly-chapters-en.tex File for Your text (replace it's contents).

projekt-20-literatura-bibliography.bib Reference list (see below).

projekt-30-prilohy-appendices-en.tex File for your appendices (replace it's contents).

projekt.tex Main project file – definitions of formal parts.

The style of literature in the template is from Ing. Radek Pyšný [**?**], whose work was improved by prof. Adam Herout, dr. Jaroslav Dytrych, dr. Petr Veigend and Mr. Karel Hanák to comply with the norm and support all frequently used types of citations. Its documentation can be found in the appendix

Aside from compilation to PDF, the Makefile also offers additional functions:

- rename files (see below),

- count standard pages,

- run a wave that adds unbreakable spaces,

- compress (zip) the result, ready to be sent to your supervisor and checked (make sure that all the files you've added are included, if not, add them manually).

Keep in mind that the wave is not perfect. You always need to check whether or not there is something inappropriate at the end of a line manually – see Online language handbook[1].

Similar rules apply also in English - see eg. article Run Ragged[2], according to which there should be no prepositions, dash or short words (2–3 letters) at the end of the lines, the two lines following each other should not end with a comma and line break should not be also in the phrases from 2-3 words.

Pay attention to page numbering! If the table of contents is 2 pages long and the second page contains only "Enclosures" and "List of enclosures" (but there is no enclosure), the page numbering is changed by 1 (table of contents and contents "mismatch"). The same thing happens if the second or third page contains only "References" and there's a chance that this can occur in other situations too. There are multiple solutions to this (from editing the table of contents, setting the page counter all the way to more sophisticated methods). **Check the page numbering before you submit your thesis!**

A.2   Recommendations for working with the template

1. **Make sure you have the latest version of template.** If you have a template from last year, there should be a newer version (updated information, fixed errors etc.) available at the faculty or study advisor web pages.

2. **Choose a language**, that you want to use for your technical report (czech, slovak or english) and consult your supervisor about your choice (unless it was agreed upon in advance). If your language of choice is not czech, set the respective template parameter in file projekt.tex (e.g.: `documentclass[english]{fitthesis}` and translate the declaration and acknowledgement to english or slovak).

3. **Rename the files.** When you extract the files, there should be a file named projekt.tex. If you compile it, it will create a PDF with technical report named projekt.pdf. If multiple students send their supervisor projekt.pdf to have it checked, they have to rename them. For that reason, it is advised to rename the file so that it contains your login and (if needed, abbreviated) work topic. Avoid using spaces, diacritic and special symbols. An appropriate name for your file can look like this: "xlogin00-Cleaning-and-extraction-of-text.tex". You can use the included Makefile to rename it:

   ```
   make rename NAME=xlogin00-Cleaning-and-extraction-of-text
   ```

4. Fill in the required information in file, that was originally named projekt.tex, that means type, year (of submission), thesis title, author's name, department (according to specification), supervisor's titles and name, abstract, keywords and other formal requirements.

---

[1]Internetová jazyková příručka http://prirucka.ujc.cas.cz/?id=880
[2]Run Raggedhttps://24ways.org/2013/run-ragged/

5. Replace the contents of thesis chapters, references and enclosures files with the contents of your technical report. Individual enclosures or thesis chapters can be saved to separate files – if you choose this approach, it is advised to comply with the file naming convention, and the number will be followed by the chapter title.

6. If you don't need enclosures, comment the respective part in projekt.tex and erase everything from the corresponding file or delete it. Don't try to come up with an aimless enclosures just to have something in that file. An appropriate enclosure can be the contents of included memory medium.

7. Delete the chapter and attachment files for a language you haven't used (with or without `-en`).

8. Assignment that you download in PDF from BUT IS (link "Thesis assignment") save to file `zadani.pdf` and enable its insertion into work by appropriate template parameter (`documentclass[zadani]{fitthesis}`) in `projekt.tex`.

9. If you don't want to print references in colour (i cannot recommend this without consulting your supervisor), you'll need to create a second PDF for printing and set the template printing parameter:
   (`documentclass[english,zadani,print]{fitthesis}`). Coloured logo must not be printed in black and white.

10. The binder templace where the thesis will be typeset can be generated in faculty IS at specification. Can be enabled for dissertation using the `cover` parameter in template.

11. Don't forget that source files and (both versions) PDF has to be on a CD or other medium included in the technical report.

A.2.1 Instructions for double-sided printing

- **It is advised to consult your supervisor about double-sided printing.**

- If you used double-sided printing for your thesis and it's thickness is smaller than the thickness of the binder, it doesn't look too good.

- Enabled using the following template parameter:
  `\documentclass[twoside]{fitthesis}`

- After printing a double-sided sheet, make sure that the canon of page construction is in the same position on both pages. Inferior printers with duplex printing unit usually cause a shift by 1–3 mm. This can be solved with some printers. Print the odd pages first, put them back into the same tray and print the even pages.

- Leave a blank page after title page, table of contents, references, list of tables, list of appendices and other lists to make sure that the following part starts on an odd page (`\cleardoublepage`).

- Check the final result thoroughly.

A.2.2 Paragraph style

Paragraphs have justified alignment and there are multiple methods for formatting them. In Czech paper literature, a paragraph indentation method is common, where each paragraph of the text have the first line of a paragraph indented by about one to two quads, that is, about two widths of the capital letter M of the base text (always about the same preselected value). In this case, the last line of the previous paragraph and the first line of the following paragraph are not separated by a vertical space. The interleaving between these lines is the same as the interleaving inside the paragraph [**?**].

Another method is indenting paragraphs, which is common for electronic typesetting and for English texts. In this method, the first line of a paragraph is not indented and a vertical space of approximately half of a line is inserted between the paragraphs. Both methods can be used in the thesis, however, the latter method is often more suitable. Methods should not be combined.

One of the above methods is set as the default in the template, the other can be selected by the template parameter "`odsaz`".

A.3   Useful tools

The following list is not a list of all useful tools. If you have experience with a certain tool, feel free to use it. However, if you don't know which tool to choose, consider the ones listed below:

MikTeX LaTeX for Windows – a distribution with simple installation and great automated package downloading. MikTeX even has it's own editor, but I highly recommend TeXstudio.

TeXstudio Portable open-source GUI for LaTeX. Ctrl+click switches between source text and PDF. Integrated spell checker[3], syntax highlighter etc. To use this tool, you need to first install MikTeX or another LaTeX distribution.

WinEdt A good combination for Windows is WinEdt + MiKTeX. WinEdt is a GUI for Windows, and if you want to use it, you need to first install MikTeX or TeX Live.

Kile Editor for KDE (Linux) desktop environment. Real-time preview. To use this tool, you need to have TeX Live and Okular installed.

JabRef Neat and simple Java program for bibliography (references) file management. No need to learn anything – provides a simple window and a form for entry editing.

InkScape Portable open-source vector graphic (SVG and PDF) editor. Excellent tool to use to create images for technical text. Difficult to master, but the results are worth it.

GIT Great tool for teamwork when it comes to projects, but can be incredibly useful even to a single author. Simple version control system, backup options and transfer between multiple computers.

---

[3]Spell checker for Czech version can be installed from https://extensions.openoffice.org/de/project/czech-dictionary-pack-ceske-slovniky-cs-cz

Overleaf Online L ATEX tool. A real-time compilation of source text that allows for simple collaboration (supervisor can continuously keep an eye on the progress made), move to a place in source file just by clicking in the PDF preview, spell checker etc. There are some limitations to what you can do if you want to use it for free (some people are comfortable with it for dissertation, others can run into it while they write a bachelor's thesis) and it is rather slow for long texts. FIT BUT has for students and employees of a license, which can be activated on `https://www.overleaf.com/edu/but`.

Note: Overleaf does not use template Makefile – to get compilation to work, you need to go to the menu and select `projekt.tex` as s Main document.

Appendix B

Writing texts in English

This chapter is taken from web pages of Jan Černocký [**?**].

A lot of people write their technical reports in English (which is good!), but they make a lot of unnecessary mistakes (which is bad!). I'm not an English export myself, but I've been using this language for a while now to write, read and even communicate – this chapter contains a handful of important things. If you want to be certain that your thesis or article is 100 % correct, your best bet is to hire a native speaker (preferably someone who is technically capable and understands what you write about . . .).

B.1 In general

- Before you jump into it head first, I suggest you read a handful of technical articles written in english and try to remember or preferably understand how you should approach writing one yourself.

- Always use a spell checking tools – built in tools in Word, or in OpenOffice. If you work on Linux, I suggest you use ISPELL. Some spell checking (I think it's the one in PSPad) are not very good and ignore a lot of mistakes.

- Use grammar checking tools. I'm not entirely sure if there is one available for Linux, but the one in Word is fairly decent and if it underlines anything with green colour, it's probably wrong. You can even copy and paste Latex source code here, fix any and all grammar errors and save it as a clean text again. If you use vim, there's a built in grammar checking tool too, and it's capable of detecting typos and errors in basic grammar. Write this in the first line of your thesis tex file:

  ```
  % vim:spelllang=en_us:spell
  ```

  (alternatively **en_gb** for OED English) *Editor's note:* There is a very good online tool Grammarly[1], with free basic version.

- Online dictionaries are good, but don't rely on them in every situation. Usually you get multiple choices and not all of them are correct for the given context.

---

[1] https://www.grammarly.com/

- You can probably figure out what the correct option is by looking each option up and seeing the context in which they're used, example given: "advantage/privilege/facility of approach". Online dictionaries give you a handful of results. Look them up one by one using google search:

```
"advantage of this approach" 1100000 hits
"privilege of this approach" 6 hits
"facility of this approach"  16 hits
```

I'm not saying it's 100 % correct, but at least you have something to go on. This can be used to find the correct connectives (e.g. "among two cases" or "between two cases"?)

B.2   SVOMPT and concord

The structure of an english sentence is SVOPMT: SUBJECT VERB OBJECT MANNER PLACE TIME and there's no other way around it. It is not a flexible structure. There are possibly exceptions in things like a theatre play, where something needs to be emphasized. Subject must be present in every single sentence, people tend to forget as some languages have a sentence structure where the subject can be implicit and not mentioned. SVOMPT applies to dependent clauses too!

```
BAD: We have shown that is faster than the other function.
GOOD: We have shown that it is faster than the other function.
```

Concord or grammatical agreement between two words in a sentence – it sounds silly, but people make countless mistakes here.

```
he has
the users have
people were
```

B.3   Articles

Articles in English are a nightmare and almost all of us fail to use them correctly. The basic rule is, that if there's a particular noun, it's preceded by "the". Definite articles must be in following phrases:

```
the first, the second, ...
the last
the most (superlatives and adverbs) ...
the whole
the following
the figure, the table.
the left, the right - on the left pannel, from the left to the right ...
```

On the contrary, there can't be an article when you're referring to a specific figure, chapter, etc.

```
in Figure 3.2
in Chapter 7
in Table 6.4
```

The use of "a" and "an" is based on the pronunciation, rather than how the word is written:

```
an HMM
an XML
a universal model
a user
```

B.4   Verbs

Passive voice can be tricky – regular verbs are usually not a problem, irregular verbs however are a common source of errors, typically

```
packet was sent (rather than send)
approach was chosen (rather than choosed)
```

. . . most of the time, the spell checker will correct it, but it's not guaranteed.

Tenses are a mess at times. If something just is in general, use present tense. If you did something, use past tense. If you got results that already exist and you just discuss them, use present tense. Try to avoid complicated tenses such as present perfect or worse past perfect if you're not 100 % sure.

```
JFA is a technique that works for everyone in speaker recognition.
We implemented it according to Kenny's recipe in \cite{Kenny}.
12000 segments from NIST SRE 2006 were processed. When compared
with a GMM baseline, the results are completely bad.
```

B.5   Sentence length and structure

- Try to write shorter sentences. If you sentence is 5 lines long, it's probably a pain to read, if it can even be done.

- Comma is a powerful tool and you should use it for your sentence structure. Use a comma to separate the initial dependent clause from the main independent clause. Sometimes it is appropriate to put a comma just before "and" (unlike other languages)!

```
In this chapter, we will investigate into ...
The first technique did not work, the second did not work as well,
and the third one also did not work.
```

B.6   The specifics of a technical text

When writing a technical text, don't use common phrases such as

```
he's
gonna
Petr's working on ...
```

and others. The only tolerated thing is "doesn't", but you can never go wrong with "does not".

Technical texts utilize passive voice a lot more than active voice:

```
BAD: In this chapter, I describe used programming languages.
GOOD: In this chapter, used programming languages are described.
```

If you want to use active voice, it's more common to use "we", even though you work alone. "I", "my", etc. are only used when you need to emphasize that you are the person of utmost importance, for example in the conclusion or when discussing "original claims" in disertation.

Common errors in words

- Pay attention to his/hers, it's not "it's" but "its"

- Image is not picture, it's figure.

- The connective is "than", not "then" – bigger than this, smaller than this . . . very common error! "Then" is used in the context of time.

Appendix C

Checklist

This checklist was taken from a template for academic work, that is available on Adam Herout's blog [**?**], based on the ideas of Igor Szöke[1], with their permission.

A big part of the safety of air transport are checklists. They have checklists for basically anything and everything, even the most cut-and-dry procedures. If a pilot can get over the tedious process of marking off every single checkbox of a procedure, you can as well. Make a checklist of your own before you submit your thesis. **Yes, really:** print it, grab a pencil and check every single item on the list. It will make your life easier — avoid unnecessary errors that can be fixed within a couple minutes — as well as others', at very least your supervisor and reviewer of your thesis.

Structure

☐ You can tell that the assignment was completed just by looking at the chapter titles as well as their structures.

☐ There is no chapter with less than four pages (except for introduction and conclusion). And if so, I discussed this with my supervisor and they gave me a green light.

Figures and charts

☐ Every single image and table was checked and their position is close to the text that references them. In other words, they're easy to find.

☐ Every single image and table has a good enough caption, to ensure that the figure makes sense on it's own, without the necessity to read the text. (There's no harm in a long caption.)

☐ If an image is taken from somewhere, it is mentioned in the caption: "Taken from [X]."

☐ Texts in all images have a font size similar to the surrounding text (neither significantly larger, nor significantly smaller).

☐ Charts and schemes are vector graphics (eg. in PDF).

☐ Screenshots don't use lossy compression (they're in PNG).

---

[1] http://blog.igor.szoke.cz/2017/04/predstartovni-priprava-letu-neni.html

☐ All images are referenced in the text.

☐ Axes in charts have their captions (name of the axis, units of measurement, values) and a grind if need be.

Equations

☐ Identifiers and their indexes in equations are single letters (except for rather uncommon cases like t_{max}).

☐ Equations are numbered.

☐ All the variables and functions that haven't been explained yet are explained below (or rarely above) the equation.

Citations

☐ **All used sources are cited.**

☐ URL addresses referencing services, projects, sources, GitHub, etc. are referenced using `\footnote{\url{...}}`.

☐ URL addresses in citations are only present, if necessary – article is cited like an article (author, title, where and when was it published), not using URL.

☐ Citations have author, title, publisher (conference title), year of publishing. If a citation does not have either of these, there is a good explanation for this special case and my supervisor agreed.

☐ If there is anything taken over from some other work in the program source code, it is properly cited therein in conformance with the license.

☐ If an essential part of the source code of the program is taken over, this is mentioned in the text of the thesis and the source is cited.

Typography

☐ No line extends past the right margin.

☐ There is no single-letter preposition at the end of a line (fixed using unbreakable space ~).

☐ Number of image, table, equation, citation is never a first item of a new line (fixed using unbreakable space ~).

☐ There is no space before a numeric reference to a footnote (like this[2], not like this [3]).

---

[2]footnote example

[3]another footnote example

Language

☐ I used spellchecker and there were no typos in the text.

☐ I had someone else read my thesis (at least one person), that knows Czech / Slovak / English well.

☐ Someone who knows English really well checked the abstract written in Czech or Slovak.

☐ No part of the text is written in the second person (you).

☐ If the first person is used (i, we), a subjective matter is being described (i decided, i designed, i focused on, i found out, etc.).

☐ There are no colloquialisms in the text.

☐ There are no *default* words in the text.

Electronic submission, software

☐ If the result is online (service, application, ...), URL is visible in introduction and conclusion.

☐ The medium contains the following mandatory items:

- source code (e.g. Matlab, C/C++, Python, ...)
- libraries necessary for compilation,
- compiled solution,
- PDF containing a technical report,
- text source code (LaTeX),

and the following optional items after consulting your supervisor:

- relevant (e.g. testing) data,
- demo video,
- poster in PDF
- ...

☐ Source codes are refactored, commented and labeled with an authorship header so that others can tell what they actually are.

☐ Any and all snippets of code taken from another sources are properly cited – differentiated using a opening and in case of multiple lines of code a closing comment. Comments contain everything that the license on web (always try to find out what the license is – for example, Stack Overflow[4] has a very strict citation policy).

---

[4] https://stackoverflow.blog/2009/06/25/attribution-required/

Submission

☐ Do I want to delay (by at most 5 years) the publication ? If so, I will submit an application (in IS) at least a month prior to the submission of the academic work, and I'll include attitude of the company that the intellectual property belongs to and needs to be protected.

☐ I have at least minimum number of standard pages (can be calculated using Makefile and by adding number of pages that images translate to or by using `https://app.fit.vut.cz/theses-checker`). If I'm just under the minimum, I consulted my supervisor about it.

☐ If I want a two-sided print, I consulted my supervisor about it and I've used correct template settings for two-sided printing. Chapters begin on odd pages.

☐ Technical report is bound in a bookbindery.

☐ Title page is followed by the specification (in other words, downloaded from IS and inserted into the template)

☐ Abstract and keywords are uploaded in IS.

    • There are no ~ characters for non-breaking spaces in the abstract and keywords in IS.

☐ PDF of thesis (with clickable links) is in IS.

☐ The print is signed.

☐ If I'm delaying the publishing, the print contains a data medium with my login written on it using a CD marker (CD marker can be borrowed in library, at Student affairs or when I'm submitting the work).

Appendix D

LaTeX for beginners

This chapter contains commonly used LaTeX packages and commands, that you might need when you're developing a thesis.

Useful packages

Students usually encounter the same issues. Some of them can be solved using the following LaTeX packages:

- `amsmath` – additional equation typesetting options,

- `float, afterpage, placeins` – image placement,

- `fancyvrb, alltt` – change the properties of Verbatim environment,

- `makecell` – additional table options,

- `pdflscape, rotating` – rotate a page by 90 degrees (for image or table),

- `hyphenat` – change how words break,

- `picture, epic, eepic` – direct image drawing.

Some packages are used in this very template (in the lower section of fitthesis.cls file). It is also advised to read the documentation for individual packages.

A table column aligned to left with a fixed width is defined as „L" in the template (used as „p").

To reference a place within text, use command `\ref{label}`. Depending on the placement of this label, it will be a number of chapter, subchapter, image, table or a similar numbered element. If you want to reference a specific page, use command `\pageref{label}`. To cite a literature reference, use command `\cite{identifier}`. To reference an equation, you can use command `\eqref{label}`.

Symbol – (dash) is used generated using two minus signs (like this: `--`) in LaTeX.

Commonly used LaTeX commands

I highly recommend you check the source text of this chapter and see how the following examples are created. The source text even contains helpful comments.

Example table:

Table D.1: Assessment table

Name		
Name	Surname	Assessment
Jan	Novák	7.5
Petr	Novák	2

Example equation:

$$\cos^3\theta = \frac{1}{4}\cos\theta + \frac{3}{4}\cos 3\theta \tag{D.1}$$

and two horizontally aligned equations:

$$3x = 6y + 12 \tag{D.2}$$
$$x = 2y + 4 \tag{D.3}$$

If you need to reference an equation from the text, you can use command `\eqref`. For example, to reference the equations above (D.1). If you want to align the equation number vertically, you can use the `aligned` environment:

$$\begin{aligned} 3x &= 6y + 12 \\ x &= 2y + 4 \end{aligned} \tag{D.4}$$

Mathematical symbols (α) and expressions can be placed even in text $\cos\pi = -1$ and can also be in a footnote[1].

Image D.1 displays a wide image comprised of multiple smaller images. Standard raster image is inserted in the same way as image D.2.

Figure D.1: **Wide image.** Image can be comprised of multiple smaller images. If you want to address the partial images from text, use packagae `subcaption`.

Sometimes it is necessary to attach a diagram that does not fit on an A4 page. Then it is possible to insert one A3 page and fold it into the thesis (so-called Engineering fold, similar

---

[1]Formula in a footnote: $\cos\pi = -1$

Figure D.2: Good text is a bad text, that has been changed countless times. You have to start somewhere.

to Z-fold, where two folds are created – face down and face up). Switching is performed as follows: `\eject \pdfpagewidth=420mm` (210mm to switch it back).

Other frequently used commands can be found above in the text, because a single practical example of correct use is better than ten pages of examples.

Appendix E

Thesis title is too long

This chapter was provided by Ing. Zaklová. If your thesis title is too long and does not look good on the first page, you can solve it using solutions provided below.

E.1   1st solution

You can modify the parameters that set the width of various title texts. To do this, uncomment and change the following in the `projekt.tex` file:

- title.length - for the main title

- sectitle.length - for the second title

- dectitle.length - for the heading above the declaration (on the page before the Table of Contents)

You have to try different settings for width and pick the correct one. This solution can cause differently sized spaces that can look bad.

E.2   2nd solution

By inserting the newline (\\) directly in the title.

This change breaks citation and heading above the declaration. To fix these problems, the following changes have to made in the `fitthesis.cls` file:

- replace \it\@title\rm. with \it{\let \\ \empty \@title}\rm.

- block

  ```
  \iftoggle{long_dectitle}{
  \mktxtl@b{18}{\@dectitlelen}{\@title}
  }{
  \mktxt@b{18}{\@title}
  }
  ```

  replace with the following block

```
\iftoggle{long_dectitle}{
\mktxtl@b{18}{\@dectitlelen}{\let \\ \empty \@title}
}{
\mktxt@b{18}{\let \\ \empty \@title}
}
```

Do not use this solution with secondary title, it would create an ugly space.
You can combine both solutions or discuss the problem with your supervisor.

Appendix F

Examples of bibliographic citations

The czplain style is based on the style created by Mr. Pyšný [**?**]. This appendix contains a set of supported type of citations with specific examples of bibliographic citations.

Examples are listed on https://www.fit.vut.cz/study/theses/citations/.en#priklady.