Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Algorithms
and Didactics

A Template
For a Computer Science Thesis

Dennis Komm

Bachelor Thesis

2025

Supervisors: Prof. Dr. First Supervisor
Prof. Dr. Second Supervisor
Dr. Third Supervisor

Abstract

This document provides an overview of how to use the `algodithesis` template, which can be used for bachelor, master, and doctoral theses. More specifically, I designed it to typeset theses written in my group *Algorithms and Didactics* at ETH, and I therefore make the first assumption that you are a student writing such a thesis. However, if anyone anywhere can find this template useful in any way, I am more than happy about that.

So feel free to use the `.cls` file in any way you want, and consider taking a look at this `.tex` file before starting. It may help you avoid some mistakes here and there. For a start, take a look at the first couple of lines of the `.tex` file that contain the basic configuration.

My second assumption is that you look at both the LaTeX code and the PDF document. Here and there, I will break the fourth wall and make important commands visible using the `\lstinline` command; like I did just now… using the `\lstinline` command.

What's more, I will try to make some more general remarks on how to typeset your thesis. Nothing here is set in stone, everything can be debated; but I am happy if you get some inspiration about what to do and what not to do when preparing your document.

Let me begin with an important message: do not touch the `\parindent` and `\parskip` lengths. I know, I know, it seems tempting; in particular if you are rather new to using LaTeX. I was there, we all were. There is no shame in it. But trust me, you will understand eventually; just leave them alone. As important: to start a new paragraph, add an empty line to the source code; like I did four times already in the text above. By all means avoid using `\\` except within tables or formulae.

Also, never use the `\noindent` command at the beginning of a paragraph. There are very few exceptions where it would make sense to use it. But as a first approximation: pretend the command does not exist.

Acknowledgements

Thank the people who helped you on your journey, both academically and personally—but don't overdo it. A three-pages Acknowledgements section for a bachelor thesis may seem awkward.

Then again, also make sure to not forget anyone. If someone invested the time to proofread parts of your thesis and provided meaningful feedback, this person deserves to be mentioned.

Contents

Chapter 1

General Remarks

This document is divided into chapters and sections. Use title case for their respective titles, which means that you should capitalize all words except for articles, prepositions, and conjunctions—unless they carry some important meaning.

Section 1.1 showcases how to typeset mathematical formulae and introduces same basic math macros. Section 1.2 contains some remarks about citations and the bibliography. Chapter 2 introduces important LaTeX environments, such as theorems and definitions in Section 2.1, floating environments like figures and tables in Section 2.2, and ordered lists in Section 2.3; finally, Section 2.4 addresses code listings.

As you can see in the paragraph above, I used `\cref` instead of `\ref` to link to numbered chapters and sections (and anything else with a label). This produces very nice clickable links. If you want to print your thesis, I strongly recommend to change the `\adthemecolor` command to `black` in the `.tex` file for obvious reasons. You can also use the resulting color `adthemecolor` in any other context, e.g., like this. This template supports all the ETH corporate design colors, which are prefixed by “ETH” such as `ETHblue`.[1]

Text can be set in *italics*, **bold**, or ***both***; it can also be *slanted*—or ***slanted and bold***; it can even be in small caps, or in **small caps and bold** or (and then I will stop) in *small caps and slanted*. To emphasize a term, use the `\emph` command, which per default sets it in *italics*. Use bold text very carefully and never underline text.

“This” is how you want to put quotation marks around something; ”this” looks bad, and "this" look even worse. In deutschen Texten, die in Deutschland publiziert werden, nehmen wir „diese“ und für die Schweiz verwenden wir wiederum «diese» Anführungszeichen.

1.1 Mathematics

In what follows, some basic mathematical macros are introduced, such as $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, or $\mathbb{R}$ for sets of numbers; if we only refer to positive numbers, we write $\mathbb{N}^+$, $\mathbb{Z}^+$, $\mathbb{Q}^+$, or $\mathbb{R}^+$ instead. $\textsc{Alg}$ and $\textsc{Alg}'$ denote some algorithms, i.e., finite descriptions of

[1] `https://ethz.ch/staffnet/en/service/communication/corporate-design/colours.html`; by the way, this is how to place a footnote marker. In particular, the “1” appears after the full stop (or comma), not before.

methods to solve given problems. Notice that there is no space between the i and e in “i.e.” whereas ein kleiner Abstand zwischen das z und B bei «z. B.» gehört.

With cost(Alg) we denote the cost of Alg, and gain(Alg′) is the gain of Alg′. $\Pr[X]$ is the probability of an event X, and $\mathbb{E}[Y]$ is the expected value of a random variable Y. The big-Oh notation is, e.g., written as $\mathcal{O}(n^2)$, $\Omega(\log n)$, and so on. Note that the logarithm function is written in upright letters, just as other functions that are not represented by single letters, such as sin, cos, and exp. So never write something like $sin(x)$. There are also some single-letter constants that are written non-italic, most notably Euler’s number e (and not e).

Equations are typeset as

$$x^2 + y^2 = z^2$$

or

$$\mathrm{e}^{i\pi} + 1 = 0 \tag{1.1}$$

if we want to refer to them, such as with (1.1). For equivalences and implications, use

$$A \implies B, \quad A \impliedby B, \quad \text{and } A \iff B\ .$$

Note that there is a small space between the formulae and the full stop. This is to make sure that no one mixes up, say,

$$\frac{1}{\textsc{Alg}'}$$

and

$$\frac{1}{\textsc{Alg}},$$

which could easily happen.

If you define a function, write, e.g.,

$$f\colon \mathbb{N}^+ \to \mathbb{R}^+$$

instead of

$$f : \mathbb{N}^+ \to \mathbb{R}^+\ ,$$

because the latter has a spacing that suggests division, such as $a : b$. To define the concrete function, you can write

$$f\colon x \mapsto 3x + 5\ ,$$

so you use ↦ instead of → in this case.

Expressions that have multiple lines are typeset as

$$\begin{aligned} 1 + 2 + \cdots + n &= \sum_{i=1}^{n} \\ &= \frac{n(n+1)}{2} \end{aligned}$$

$$= \frac{n^2+n}{2} \,,$$

and there is no double backslash in the last line. If you want to refer to single lines, you can assign labels to them by

$$\begin{aligned}\binom{n}{k} &= \frac{n!}{k!\cdot(n-k)!}\\ &= \frac{n!}{(n-n+k)!\cdot(n-k)!}\\ &= \frac{n!}{(n-(n-k))!\cdot(n-k)!} \qquad (1.2)\\ &= \binom{n}{n-k}\end{aligned}$$

and refer to the third line by (1.2). Multiple equivalence operations are typeset as

$$\begin{aligned} && (\sin\alpha)^2+(\cos\alpha)^2 &= 1\\ \iff && (\sin\alpha)^2 &= 1-(\cos\alpha)^2\\ \iff && \sin\alpha &= \pm\sqrt{1-(\cos\alpha)^2}\,.\end{aligned}$$

Finally, if for some reason you want to have, say, a formula inside bold text, make sure to use `\boldmath` in order to avoid ugly texts like "**Let** $x = y^3$ **be a cubic number.**" and have instead the much prettier "**Let $\boldsymbol{x = y^3}$ be a cubic number.**" Be warned, however: if the boldface text is, say, a section title, automatic crawling may unintentionally make the commands part of the text.

1.2 Bibliography and References

If you use a bibliography that exceeds a few items, I recommend you use BibTeX to manage them. In this example `.tex` file, I don't, because I only want to demonstrate a few points that are independent of the system used to manage the references.

When citing something, you should avoid using the reference as a word, i.e., writing something like "see [7]" or "as shown in [1]." It is not an absolute faux pas to do so (you will actually see it quite frequently I am afraid), but rather treat the brackets as what they are, and write "as shown by Turing [1]." If a paper is written by two authors, write both their names; if there are more, only mention the first one, e.g., Rivest et al. [4]. Here, you should only use the surnames of the authors, so avoid "as demonstrated by Donald Knuth [2]" or even "as demonstrated by Prof. Donald Knuth [2]."

However, I personally think that it is a very nice gesture to spell out the full names in the bibliography at the end of the document. This is not the default setting with most publishers, unfortunately; but I would like to encourage you to do so—well, if it is up to you. It shows that you care about the people who built the foundation of your work. Likewise, if it is within your power, add the DOI of each paper and make it a clickable link. This makes reading (and reviewing) your thesis much more convenient and enjoyable. (If you cannot find a DOI, you may want to try to find another authentic online resource.)

Now to something different: there is a difference between a plain dot and a full stop in that TEX puts more space behind the latter. If a dot is not supposed to be full stop, this is indicated by an explicit inter-word space. I did so above behind the "et al." to tell TEX that the sentence continues afterwards and thus that the space behind the dot should be smaller. (As you can see in the `.tex` file, I use two white spaces to separate sentences when writing plain text.)

Chapter 2

Environments

You should try to avoid having two headlines directly below of each other; there should be some text in between. Of course, this is not a strict rule, but it looks nicer this way, and it is a good opportunity to give the reader a brief high-level description of what will be written on the next couple of pages.

What's more, note that new chapters always start on odd pages.

2.1 Definitions and Theorems

Definitions, theorems, lemmata, etc. (again the explicit inter-word space just mentioned at the end of Section 1.2) are typeset using the corresponding environments.

Definition 2.1 (Algorithm). *In the parentheses we add the term that is defined. In many cases you may also want to add a reference such as in the following example.*

Definition 2.2 (Algorithm [7]). *There you go. By no means, however, put a reference without anything else there.*

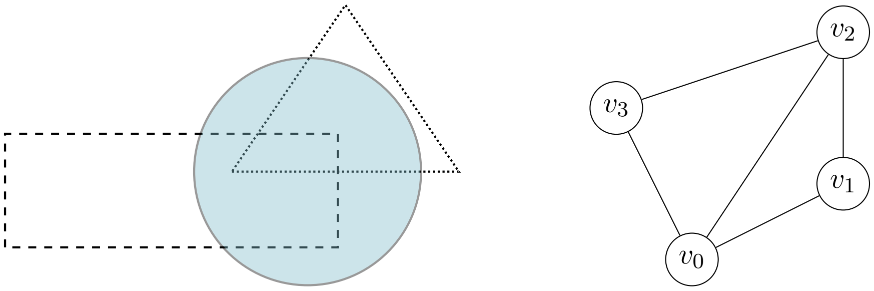
Definition 2.3 ([7]). *This looks very ugly, and should be avoided. If you stick to the above treating "the brackets as what they are" [3], it also does not make sense to open parentheses to only open brackets within: the parentheses should contain something else as well.*

In what follows, you will see how to typeset a theorem and its proof. Note that the qed-symbol is placed at the end automatically. This also works if your proof ends with an environment as in that of Lemma 2.1. Just make sure there is no empty line right before the `\end{proof}`.

Theorem 2.1. *There are infinitely many primes.*

Proof. Towards contradiction, suppose this is not the case. There are therefore, say, $k \in \mathbb{N}$ primes $p_1, \dots, p_k$. This is a direct contradiction as $p \coloneqq p_1 \cdot \ldots \cdot p_k + 1$ gives us a $(k+1)$th prime as it cannot be divided by any of the other k primes. □

Notice that in the proof of Theorem 2.1, I used the special symbol "$\coloneqq$" (the macro `\da`) for "defined as" and not the straightforward "$:=$" although I admit that



(a) Geometric primitives

(b) An undirected graph

Figure 2.1. A figure consisting of two subfigures

probably no one will see the difference. But if you take a very close look, you will see how awkwardly aligned the colon and equal sign are in ":=".

Next, there is a corollary instead of a theorem. Since it has no proof, the qed-symbol is manually placed at its end.

Corollary 2.1. *As a consequence of Theorem 2.1, there is no largest prime.* □

This template contains a special environment for case distinctions, which also allows you to define subcases.

Lemma 2.1. *For every $n \in \mathbb{N}$, $n \cdot (n+1)$ is even.*

Proof. We distinguish two cases, which we prove separately.

Case 1. Suppose n is odd. Then $n+1$ is even and therefore divisible by 2. Consequently, $n(n+1)$ is divisible by 2 and thus even as well by definition.

Case 2. Suppose n is even. In this case, n is divisible by 2, which implies that $n(n+1)$ is again even. □

2.2 Floating Environments

There are multiple ways to draw figures, but you should be careful to choose one that is able create vector graphics. Nothing is more frustrating than a pixelated image in an otherwise neatly produced PDF document. I strongly recommend Ti*k*Z [6] as it nicely interacts with LaTeX.

Figures are put into specific environments, which "float" in the text. This means that they do not have to appear at the exact position where they are referenced. Therefore, there should be no sentences like "Consider the following figure:" in your document. Always refer to the label of the figure and write "Consider Figure 2.1." or similar instead; there may be times where a drawing is inside the text, even within a line, but then it should not be put into a figure environment.

If possible, figures are displayed on the top of the page on which they are referred to for the first time. You should always do it this way, and *not* place figures at absolute positions.

Table 2.1. Different functions

n	10	100	200	1 000
$\log_{10} n$	1	2	≈ 2.30	3
$3n$	30	300	600	3 000
n^2	100	10 000	40 000	100 000
$n^3 + n/2$	1 005	1 000 050	8 000 100	100 000 500

This placement on the top should not be changed, although it seems tempting to place figures at the position where they are referenced by using the parameter `!h`. Don't do this. Just as with the indentation of paragraphs mentioned in the abstract, you will get used to it; and you will find that it makes your document look much more consistent and neat.

Note that the `\label` command associated with a figure is placed after the corresponding `\caption` command. Moreover, in the produced PDF, the caption is placed below the figure.

Figure 2.1 is at the top of the previous page and consists of two subfigures, which also have their own labels—namely, Figures 2.1(a) and 2.1(b). As mentioned above, the figure is placed on the page where it is first referenced—which is indeed the previous page.

Tables should be treated similar to figures, and therefore also float and ideally be placed on the top of pages. In contrast to figures, however, the captions of tables are displayed above the tables and not below. Again, the `\label` is placed after the `\caption`. An example is shown in Table 2.1.

Although you will find many counterexamples, vertical lines should be avoided when typesetting tables; they should only contain horizontal lines, in particular to frame them.

Note how I inserted small spaces behind blocks of three consecutive digits when writing large number, e.g., I wrote 10 000 instead of 10000, to make it easier to parse. Besides that, I aligned the numbers to the right, what makes it easier to compare them—in particular when also using the spacing.

2.3 Ordered Lists

The standard way of typesetting an ordered list (i.e., a list of enumerated items) is using the default `enumerate` environment. This template, however, also comes with two special environments `enumerateroman` and `enumeratealph` that allow

(i) you to enumerate items

(ii) using roman numbers, for example.

The nice thing is that you can also refer to the single items by (i). This may come in handy in particular if you want to refer to a numbered part of a definition or theorem.

Code Listing 2.1. Binary search as an iterative Python function

```
 1 def binsearch(data, searched):
 2     left = 0
 3     right = len(data) - 1
 4     while left <= right:
 5         current = (left + right) // 2
 6         if data[current] == searched:
 7             return current
 8         elif data[current] > searched:
 9             right = current - 1
10         else:
11             left = current + 1
12     return -1
```

2.4 Code Listings

If you want to typeset code, this can be done in special environment. This template so far supports Python code out of the box (also highlighting the keyword `repeat` for TigerJython code), but this can be easily extended to any other language of your choice.

In general, I personally would treat code listings, in particular if they are significantly large, less like formulae or lists, and more like figures and tables, and consequently recommend that you make them float and have both a caption and label.

Note that setting the caption and label works differently for listings; both are specified as optional arguments of the corresponding environment.

This is done with Listing 2.1 on the top of this page, which also allows you to reference single line number, e.g., by line 4.

If you only write a few lines of code, you may want to think about not having it float, but place it right where it is referred to:

```
def factorial(n):
    if n == 1:
        return 1
    else:
        return n * factorial(n-1)
```

This is in particular a good idea if small code fragments take turns with blocks of text. In that case, it may make the reading more convenient since the reader does not have to jump between lines (or even) pages.

Chapter 3

Further Reading

As mentioned above, there is not one correct way to do things. But it is good to put some thoughts into it, and have a good idea about why certain things are somewhat the default.

If you want to do a deep dive and also learn how TEX operates internally, the probably best resource is Knuth's TEXbook [2]. A classical resource that is worth looking into, but that has no focus on scientific writing, is *The Elements of Style* [5] by Strunk and White.

You will find contradictions in this literature both between different resources, and this template. At the end of the day, find your own style. But invest some time and make it count; you will be not be writing many theses.

Bibliography

[1] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing (STOC 1971)*, pages 151–158. Association for Computing Machinery, 1971. https://doi.org/10.1145/800157.805047.

[2] Donald E. Knuth. *The TEXbook.* American Mathematical Society and Addison–Wesley Publishing Company, 1986. https://ctan.org/pkg/texbook?lang=en.

[3] Dennis Komm. *A Template – For a Computer Science Thesis.* 2025. https://people.inf.ethz.ch/dkomm/.

[4] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM* 21(2):120–126, 1978. https://doi.org/10.1145/359340.359342.

[5] William Strunk Jr. and E. B. White. *The Elements of Style.* 4th edition, Pearson, 1999.

[6] Till Tantau. The Ti*k*Z and PGF Packages. Unofficial HTML version maintained by Dominik Peters. https://tikz.dev.

[7] Alan M. Turing. On computable numbers, with an application to the Entscheidungsproblem. In *Proceedings of the London Mathematical Society*, 42(2):230–265, 1936. https://doi.org/10.1112/plms/s2-42.1.230.