

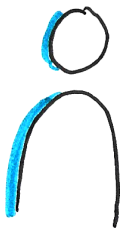
# How to Write a (Seminar) Report?

Liat Cohen

University of Basel

# Let's Go!

# Goals



You

practice reading and writing  
scientific literature



Other seminar participants

- Who is your audience?
- What do they already know?
- What should they learn?

# Source Material

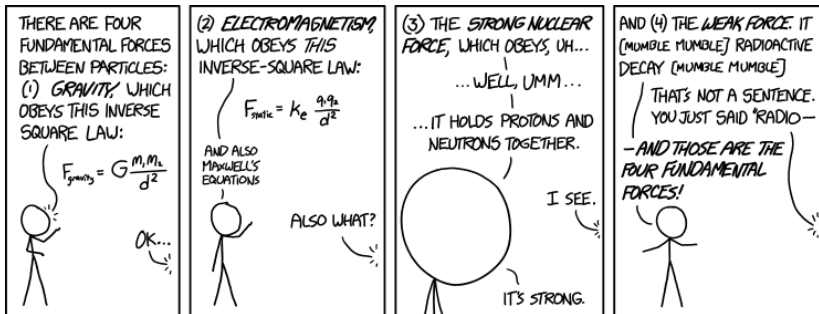
## Finding source material

- References in existing material
- <https://scholar.google.com> (demo)
- References in Wikipedia articles (in the end of the article)
- Library
- Ask your supervisor for help!

## Articles behind a paywall?

- Authors' homepages
- University subscriptions
- Library

# Understanding the Source Material



[www.xkcd.com](http://www.xkcd.com)

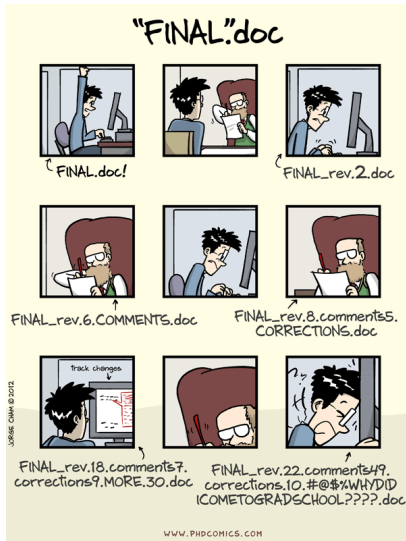
- Do not ignore complex details.
- We are happy to help.

# Start Early



- Writing is an **iterative process**.
- **Don't wait** until you read everything.
- Get feedback on drafts.
- Be prepared to make **revisions**.

# Version Control Systems



- Use **version control** for files you create (tex).
- **Ignore** automatically created files (pdf, log, aux, ...).
- **Overleaf**, Online LaTeX Editor or repositories on **Bitbucket** or **Github**

# Structure



# A Common Structure

- Abstract
- Introduction
- Background
- Main Part(s)
- Related Work
- Conclusion
- References

# Abstract I

## Sorting Algorithms

Hans Meier

Seminar on Algorithms and Data Structures  
University of Basel  
HS 20XX

### Abstract

A *sorting algorithm* orders the elements of a list according to a given total order relation. We explain three different such algorithms, namely *merge sort*, *heap sort* and *quick sort* and analyse their time and space complexity. An empirical evaluation illustrates in which scenarios these algorithms have their strenghts and weaknesses.

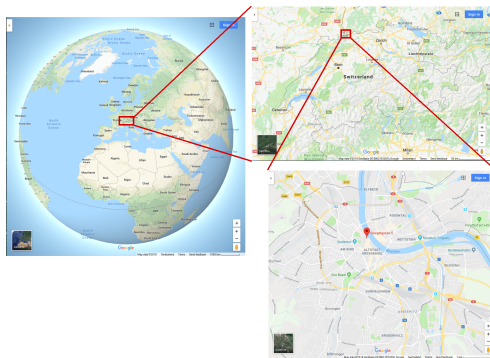
## 1 Introduction

# Abstract II



- very short summary of the report's content
- high-level, no details, no references
- potential readers decide whether to read on
- expectation management
- in  $\text{\LaTeX}$ : `\begin{abstract}...\end{abstract}`

# Introduction I



- context
- (gap in previous approaches)
- motivation, why the topic is interesting
- high-level description of the topic

# Introduction II

## ■ structure of the report

*We present an extension of the LM-Cut heuristic that preserves both admissibility and dominance over the maximum heuristic. For this purpose we introduce context splitting as a new general technique which allows us to split up actions in a tasks to distinguish different scenarios of their application. We show how context splitting can be made useful for the extension of the LM-Cut heuristic. After proving the desired theoretical properties of the heuristic, we also evaluate its performance empirically.*

[Röger et al., ECAI 2014]

# Background

- introduces basic terminology and notation
- [foundation](#) of main parts, not a goal in itself
- often general or known definitions or previous work
- makes report [self-contained](#).
- title does not have to be “Background”
  - SAS<sup>+</sup> Planning,  $\mu$ -recursive Functions, Turing Machines, ...

## Background (style)

- formal language
  - clear, easy to read, unique interpretation
  - not too complicated
  - not too colloquially

### More details

Zobel, J. (2015). *Writing for computer science*. Springer.

## Examples for good and bad style

### No unique interpretation

If some nodes get me from A to B, I'll call them a path.

### Too complicated

A path is  $\pi = \langle e_i \mid \forall i : 1 \leq i \leq n \rangle$ ,  $e_i = \langle v_{i,1}, v_{i,2} \rangle \in E$ ,  
 $\forall i : 1 \leq i \leq n$ , and  $v_{i,2} = v_{i+1,1} \forall i : 1 \leq i \leq n - 1$ .

### Unique interpretation and easy to read

A path is a sequence of nodes such that there is an edge between each pair of subsequent nodes.



# Main Parts

- main part of your report
- structure depends on the topic

## Related Work

- short description of other approaches for the same problem or similar problems
- focus on core ideas
- sometimes also directly after introduction

# Conclusion

- short summary of the main results
- **do not repeat** abstract or introduction
- often ends with open questions  
or discusses how work can be continued

# References

- list of used literature (and other sources)
- **complete** and **consistent**
  - don't use "Proceedings of the Xth Conference on Blabla" for one conference and "Proc. ACRONYM 2000" for another
  - or even worse: the same conference
- use bibtex, biblatex, . . .
- read the output of these tools
  - warnings for incomplete entries

# Citations

# Citing

- “Meier and Huber (2013) have shown...”
- “Für das  $n^2$ -Puzzle ist es NP-schwer eine kürzeste Lösung zu finden (Ratner und Warmuth 1986).”
- **Theorem 1** (Murphy’s law, Sack 1952).  
*Anything that can possibly go wrong, does.*
- **not** “(Meier and Huber 2013) have shown...”
- **not** “In (Ratner und Warmuth 1986) ...”

# Bibtex

```
@Book{hofstadter-1979,  
  author = "Douglas R. Hofstadter",  
  title = "Gödel, Escher, Bach:  
          an eternal golden braid",  
  year = "1979",  
  publisher = "Basic books"}
```

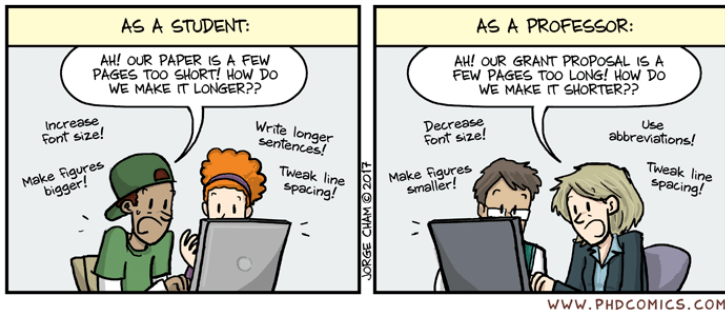
(demo)

# FAQ



# How can I fill the pages?

## PAGE LIMITS



- explain things in more detail
- explain more things
- use more examples

# What are common mistakes?

- Using terms/notation before they are introduced
- Only translating/paraphrasing an original text
- Colloquial or ambiguous language
- “The authors wrote this in the best way possible. How should I write this in a different way?”
  - get a deeper understanding of the material
  - read alternative sources
  - do not read source while writing

## Where can I get further information?

### Good example

We uploaded a good example from last year to the course website.

### Great talk by Simon Peyton Jones

<https://www.microsoft.com/en-us/research/academic-program/write-great-research-paper/>

# Questions

