



UNIVERSITÄT ZUM BEISPIEL
INSTITUT FÜR BEISPIELE

Wie schreibe ich eine Masterarbeit?

How to Write a Master Thesis?

Masterarbeit

im Rahmen des Studiengangs
Informatik
der Universität zum Beispiel

vorgelegt von
Max Mustermann

ausgegeben und betreut von
Prof. Dr. Erika Musterfrau

mit Unterstützung von
Lieschen Müller

Die Arbeit ist im Rahmen einer Tätigkeit bei der Firma Muster GmbH entstanden.

Musterhausen, den 15. Juli 2016

Erklärung

Ich erkläre hiermit an Eides statt, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

(Max Mustermann)
Musterhausen, den 15. Juli 2016

Kurzfassung Die Kurzfassung sollte nicht länger als einige Absätze sein. Für die deutsche Kurzfassung muss eine englische Version existieren, die eine genaue Übersetzung der deutschen Fassung sein muss.

Abstract The abstract should not be longer than some paragraphs. There must be an English translation of the the German abstract, which has to be the exact translation of the German version.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Verwandte Arbeiten	1
1.2. Aufbau der Arbeit	1
2. Grundlagen	3
2.1. Zielgruppe	3
2.2. Umfang	3
2.3. Quellen	3
3. Konzept	5
3.1. Quellen	5
3.2. Tabellen	6
3.3. Abbildungen und Diagramme	6
3.4. Quelltext	6
3.4.1. Quelltext mit automatischer Nummerierung	8
3.5. Pseudocode	8
3.6. Formeln mit ε	8
3.7. Theoreme	9
3.8. Anführungszeichen	9
4. Evaluierung	11
4.1. Implementierungen	11
5. Zusammenfassung und Ausblick	13
A. Anhang	15
A.1. Abschnitt des Anhangs	15

Aufgabenverzeichnis

Die Abschnitte dieses Kapitels sollten natürlich nicht so in die Arbeit übernommen werden.	5
Notizen an einen selbst oder den Betreuer der Arbeit sind während der Arbeit sehr nützlich. Für die finale Version können diese Todo-Notes dann komplett aufgeblendet werden.	5

1. Einleitung

Die Einleitung führt zum eigentlichen Thema dieser Arbeit hin. Dabei wird ein großer Bogen gespannt, in dem die Relevanz und der Kontext der untersuchten Thematik deutlich wird. Grundlegende Begriffe aus dem Titel und der Kurzfassung sollten aufgegriffen und definiert werden. Unterstützend können Zitate herangezogen werden, die der Arbeit einen Rahmen geben.

1.1. Verwandte Arbeiten

Eine wichtiger Abschnitt der Einleitung stellt einen Überblick über verwandte Arbeiten dar. Was wurde bereits in der Literatur untersucht und ist *nicht* Thema dieser Arbeit?

1.2. Aufbau der Arbeit

Neben dieser Einleitung und der Zusammenfassung am Ende gliedert sich diese Arbeit in die folgenden drei Kapitel.

Kapitel 2 beschreibt die für diese Arbeit benötigten Grundlagen. In diesem Kapitel werden . . . , . . . und . . . eingeführt, da diese für die folgenden Kapitel dringend benötigt werden.

Kapitel 3 stellt das eigentliche Konzept vor. Dabei handelt es sich um ein Konzept zur Verbesserung der Welt. Das Kapitel gliedert sich daher in einen globalen und einen lokalen Ansatz, wie die Welt zum Besseren beeinflusst werden kann.

Kapitel 4 beinhaltet eine Evaluation des Konzeptes aus dem vorherigen Kapitel. Anhand von Simulationen wird in diesem Kapitel untersucht, wie die Welt durch konkrete Maßnahmen deutlich verbessert werden kann.

2. Grundlagen

Dieses Kapitel beschreibt alle für die Arbeit notwendigen Grundlagen.

2.1. Zielgruppe

Die Zielgruppe einer Abschlussarbeit sind natürlich in erster Linie die Gutachter, die am Ende die Arbeit lesen und bewerten. Als Richtlinie, welches Wissen beim Leser einer solchen Arbeit vorausgesetzt werden kann, sollte man sich allerdings einen Kommilitonen des gleichen Studiengangs vorstellen, der in einem anderen Fachbereich seine Abschlussarbeit schreibt. Für diesen sollten wenigstens alle wesentlichen Definitionen enthalten sein.

2.2. Umfang

Eine Abschlussarbeit ist allerdings kein Lehrbuch. Entsprechend ist vor allem die Korrektheit wichtig. Zu umfangreiche Beispiele für in der Literatur bereits zur Genüge untersuchte Grundlagen sollten vermieden werden, um den Leser nicht zu langweilen.

2.3. Quellen

Ein wesentliches Charakteristikum von Grundlagen ist, dass diese nicht vom Autor der Arbeit erfunden wurden. Entsprechend ist gerade in den Grundlagen darauf zu achten, ausreichend Quellen anzugeben. Eine gute Regel ist dabei, immer den Erfinder bzw. das erste Auftauchen eines Konzeptes in der Literatur und mindestens eine gut verständliche neuere Quelle anzugeben. In den meisten Fällen hat sich seit der Einführung einer neuen Idee einiges getan, so dass neuere Quellen meistens einen besseren Einstieg in das Thema bieten. Es gehört sich aber, den Erfinder immer mit zu zitieren, da dieser die initiale Idee hatte.

3. Konzept

In diesem Kapitel wird die eigentliche Erkenntnis dieser Arbeit beschrieben. Der Aufbau dieses Kapitels hängt stark vom Thema der Arbeit ab. Die in dieser Vorlage vorgeschlagenen Kapitel sind auch nur als Vorschlag und auf keinen Fall als verbindlich zu verstehen.

Die folgenden Abschnitte dieses Kapitels enthalten Beispiele für die diversen Inhaltelemente einer Arbeit.

Notizen an einen selbst oder den Betreuer der Arbeit sind während der Arbeit sehr nützlich. Für die finale Version können diese Todo-Notes dann komplett aufgeblendet werden.

Die Abschnitte dieses Kapitels sollten natürlich nicht so in die Arbeit übernommen werden.

3.1. Quellen

Quellen sind wichtig für gutes wissenschaftliches Arbeiten. Eine Quelle kann dabei zum Beispiel

- ein Beitrag in einer Zeitschrift [MJG⁺12],
- ein Beitrag in einem Sammlungsband [Moo56],
- ein Buch [OSV08],
- ein Beitrag im Berichtsband einer Konferenz [LS07],
- ein technischer Bericht [ZM10],
- eine Dissertation [Leu02],
- eine Abschlussarbeit [Sch12],
- ein (noch) nicht veröffentlichter Artikel [Ros07] oder
- ein Artikel auf einer Website [Spi09] sein.

Dabei ist zu beachten, dass nicht veröffentlichte Artikel und insbesondere Webseiten nur in Ausnahmefällen gute Quellen sind, da diese nicht durch Fachleute begutachtet wurden.

Im Bereich der Informatik können Quellenangaben im BibTeX-Format direkt der `dblp`¹ entnommen werden.

¹zum Beispiel <http://dblp.uni-trier.de>

3. Konzept

Jahr	Prozessor	MHz
1975	6502 (C64)	1
1985	80386	16
2005	Pentium 4	2 800
2030	Phoenix 3	7 320 000
2050	...	
2070	...	

Tabelle 3.1.: Rechengeschwindigkeit von Computern. Inhaltlich vollkommen egal, ist dies doch ein sehr schönes Beispiel für eine Tabelle.

3.2. Tabellen

In Tabelle 3.1 sehen wir ein Beispiel für eine Tabelle.

3.3. Abbildungen und Diagramme

In Abbildung 3.1 auf der nächsten Seite sehen wir ein Beispiel für eine Abbildung, die aus einer externen Grafik geladen wurde. In Abbildung 3.2 auf der nächsten Seite sehen wir ein Beispiel für eine Abbildung, die in \LaTeX generiert wurde.

3.4. Quelltext

Quelltext sollte in Abschlussarbeiten nur äußerst sparsam eingesetzt werden. Wichtig ist insbesondere, dass Quelltextauszüge sorgsam ausgewählt und gut erklärt werden.

```
public class Main {  
    // Hello Word in Java  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

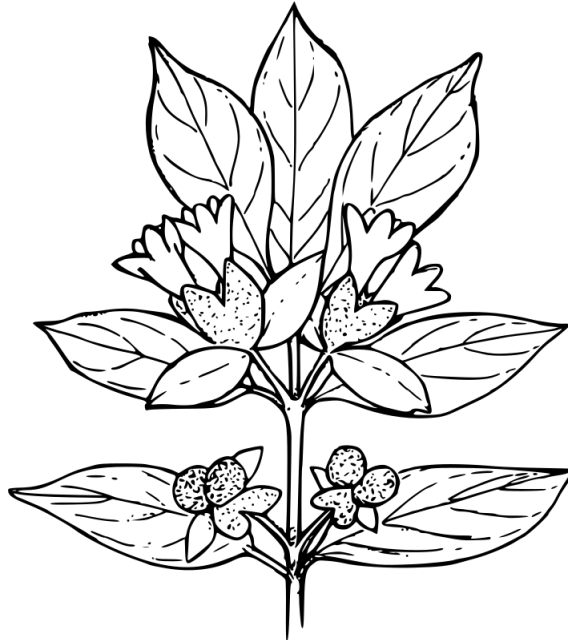


Abbildung 3.1.: Lange Version der Beschreibung, die direkt unter der Abbildung gesetzt wird. Es ist wichtig, für jede Abbildung eine umfangreiche Beschreibung anzugeben, da der Leser beim ersten Durchblättern der Arbeit vor allem an den Abbildungen hängen bleibt.

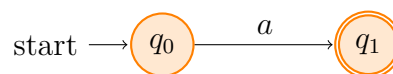


Abbildung 3.2.: Graph des Büchi-Automaten \hat{A} . Der Zustand q_1 hat dabei keine ausgehende Kante. Der Zustand ist trotzdem akzeptierend, da beide enthaltenen Zustände von \hat{A} akzeptierend sind. Die naive Anwendung des Leerheitstests auf alternierenden Büchi-Automaten liefert in diesem Fall also zu viele akzeptierende Zustände.

3.4.1. Quelltext mit automatischer Nummerierung

Manchmal möchte man Quelltext eher als Abbildung und nicht als Fließtext behandeln. In diesem Fall soll der Quelltext eine Bildunterschrift und eine automatische Nummerierung erhalten. Die automatische Nummerierung kann dann natürlich auch in Referenzen verwendet werden: In Quelltext 3.1 haben wir Java-Quelltext.

```
public class AnotherClass {  
    private int number = 0;  
    public void add() {  
        this.number++;  
    }  
}
```

Quelltext 3.1: Ich bin die Bildunterschrift des Quelltextes

Wenn man Quelltext mit Bildunterschrift setzt, muss man darauf achten, dass der Quelltext nach wie vor nicht als Fließumgebung behandelt wird. Entsprechend kann es passieren, dass der Quelltext über zwei Seiten hinweg gesetzt wird. Während das normalerweise nicht stört, kann dieser Umstand in Zusammenhang mit einer Bildunterschrift den Leser irritieren.

3.5. Pseudocode

Um Algorithmen zu erklären ist Pseudocode viel besser geeignet als Quelltext. Im Pseudocode kann man alles unwichtige weglassen und sich auf die mathematische Modellierung des Algorithmus konzentrieren. So kann die Struktur des Verfahrens unabhängig von Implementierungsdetails des jeweiligen Frameworks erklärt werden.

```
// Schleife von 1 bis 5  
for  $i \leftarrow 1$  to 5 do  
    while  $S[i] \neq S[S[i]]$  do  
         $S[i] \leftarrow S[S[i]]$ 
```

3.6. Formeln mit ε

Das ε in der Überschrift dieses Abschnitts ist ein Beispiel für ein mathematisches Symbol, das in den PDF-Lesezeichen als reiner Text gesetzt wird. Siehe `macros.tex`. In dieser Datei wird auch $n \in \mathbb{N}$ definiert.

Wir wissen aus der Analysis, dass

$$f(x) = x^2 + px + q$$

Nullstellen bei

$$x_1 = -\frac{p}{2} + \sqrt{\frac{p^2}{4} - q} \text{ und}$$

$$x_2 = -\frac{p}{2} - \sqrt{\frac{p^2}{4} - q}$$

hat.

3.7. Theoreme

Definition 3.1 (Definition). Eine *Definition* ist die Bestimmung eines Begriffs oder eines mathematischen Zusammenhangs.

Lemma 3.2 (Unwichtiger Hilfssatz). *Der Satz ...*

Beweis. ... und sein Beweis. □

Theorem 3.3 (Wichtiger Satz). *Ein wichtiger Satz.*

Beweis. Der Beweis folgt unter Verwendung der Erkenntnisse aus Lemma 3.2. □

Korollar 3.4 (Ein Geschenk). *Eine unmittelbare Folgerung.*

Beweis. Die Folgerung folgt unmittelbar aus Theorem 3.3. □

Beispiel 3.5. Ein Beispiel.

Bemerkung 3.6. Beispiele und Bemerkungen werden nicht in das Verzeichnis der Theoreme und Definitionen aufgenommen.

3.8. Anführungszeichen

Malte sagt: »Fritz hat gesagt: ›Man kann wörtliche Rede verschachteln.‹«

Anführungszeichen werden nur für wörtliche Rede oder direkt übernommene kurze Zitate verwendet. Für die Hervorhebung von neu eingeführten Fachbegriffen eignet sich *kursiver Satz* besser.

4. Evaluierung

In der Evaluierung wird das Ergebnis dieser Arbeit bewertet. Eine praktische Evaluation eines neuen Algorithmus kann zum Beispiel durch eine Implementierung geschehen. Je nach Thema der Arbeit kann sich natürlich auch die gesamte Arbeit eher im praktischen Bereich mit einer Implementierung beschäftigen. In diesem Fall gilt es am Ende der Arbeit insbesondere die Implementierung selber zu evaluieren. Wesentliche Fragen dabei können sein:

- Was funktioniert jetzt besser als vor meiner Arbeit?
- Wie kann das praktisch eingesetzt werden?
- Was sagen potenzielle Anwender zu meiner Lösung?

4.1. Implementierungen

Wenn Implementierungen umfangreich beschrieben werden, ist darauf zu achten, den richtigen Mittelweg zwischen einer zu detaillierten und zu oberflächlichen Beschreibung zu finden. Eine Beschreibung aller Details der Implementierung ist in der Regel zu detailliert, da die primäre Zielgruppe einer Abschlussarbeit sich nicht im Detail in den geschriebenen Quelltext einarbeiten will. Die Beschreibung sollte aber durchaus alle wesentlichen Konzepte der Implementierung enthalten. Gerade bei einer Abschlussarbeit am Institut für Softwaretechnik und Programmiersprachen lohnt es sich, auf die eingesetzten Techniken und Programmiersprachen einzugehen. Ich würde in einer solchen Beschreibung auch einige unterstützende Diagramme erwarten.

5. Zusammenfassung und Ausblick

Die Zusammenfassung greift die in der Einleitung angerissenen Bereiche wieder auf und erläutert, zu welchen Ergebnissen diese Arbeit kommt. Dabei wird insbesondere auf die neuen Erkenntnisse und den Nutzen der Arbeit eingegangen.

Im anschließenden Ausblick werden mögliche nächste Schritte aufgezählt, um die Forschung an diesem Thema weiter voranzubringen. Hier darf man sich nicht scheuen, klar zu benennen, was im Rahmen dieser Arbeit nicht bearbeitet werden konnte und wo noch weitere Arbeit notwendig ist.

A. Anhang

Dieser Anhang enthält tiefergehende Informationen, die nicht zur eigentlichen Arbeit gehören.

A.1. Abschnitt des Anhangs

In den meisten Fällen wird kein Anhang benötigt, da sich selten Informationen ansammeln, die nicht zum eigentlichen Inhalt der Arbeit gehören. Vollständige Quelltextlisting haben in ausgedruckter Form keinen Wert und gehören daher weder in die Arbeit noch in den Anhang. Darüber hinaus gehören Abbildungen bzw. Diagramme, auf die im Text der Arbeit verwiesen wird, auf keinen Fall in den Anhang.

Abbildungsverzeichnis

3.1. Kurzfassung der Beschreibung für das Abbildungsverzeichnis	7
3.2. Graph des Büchi-Automaten \hat{A}	7

Tabellenverzeichnis

3.1. Rechengeschwindigkeit von Computern 6

Definitions- und Theoremverzeichnis

3.1. Definition (Definition)	9
3.2. Lemma (Unwichtiger Hilfssatz)	9
3.3. Theorem (Wichtiger Satz)	9
3.4. Korollar (Ein Geschenk)	9

Quelltextverzeichnis

3.1. Ich bin die Bildunterschrift des Quelltextes	8
---	---

Abkürzungsverzeichnis

ABA	alternierender Büchi-Automat, engl. <i>alternating Büchi automaton</i>
AFA	alternierender endlicher Automat, engl. <i>alternating finite automaton</i>
BA	Büchi-Automat, engl. <i>Büchi automaton</i>
BNF	Normalform kontextfreier Grammatiken, engl. <i>Backus–Naur form</i>
DFA	endlicher Automat, engl. <i>deterministic finite automaton</i>

Literaturverzeichnis

- [Leu02] LEUCKER, Martin: *Logics for Mazurkiewicz Traces*, RWTH Aachen, Dissertation, 2002
- [LS07] LEUCKER, Martin ; SÁNCHEZ, César: Regular Linear Temporal Logic. In: JONES, C. B. (Hrsg.) ; LIU, Z. (Hrsg.) ; WOODCOCK, J. (Hrsg.): *Proceedings of the 4th International Colloquium on Theoretical Aspects of Computing (ICTAC)* Bd. 4711, Springer, 2007 (Lecture Notes in Computer Science), S. 291–305
- [MJG⁺12] MEREDITH, Patrick O. ; JIN, Dongyun ; GRIFFITH, Dennis ; CHEN, Feng ; ROŞU, Grigore: An Overview of the MOP Runtime Verification Framework. In: *International Journal on Software Tools for Technology Transfer (STTT)* 14 (2012), Nr. 3, S. 249–289
- [Moo56] MOORE, Edward F.: Gedanken-Experiments on Sequential Machines. In: SHANNON, Claude (Hrsg.) ; MCCARTHY, John (Hrsg.): *Automata Studies*. Princeton University Press, 1956, S. 129–153
- [OSV08] ODESKY, Martin ; SPOON, Lex ; VENNERS, Bill: *Programming in Scala: A Comprehensive Step-by-step Guide*. 1. Auflage. Artima Incorporation, 2008
- [Roş07] ROŞU, Grigore: *A Monitor Synthesis Algorithm for Past LTL*. Frühling 2007. – Vorlesung am Formal Systems Laboratory, Department of Computer Science at the University of Illinois at Urbana-Champaign
- [Sch12] SCHMITZ, Malte: *Transformation von regulärer Linearzeit-Temporallogik zu Paritätsautomaten*, Universität zu Lübeck, Bachelorarbeit, 2012
- [Spi09] SPIEWAK, Daniel: *The Magic Behind Parser Combinators*. <http://www.codecommit.com/blog/scala/the-magic-behind-parser-combinators>, 2009. – [Online; Zugriff am 03.03.2014]
- [ZM10] ZIEGLER, Stephan ; MÜLLER, Anne: Eingebettete Systeme – Ein strategisches Wachstumsfeld für Deutschland: Anwendungsbeispiele, Zahlen und Trends / Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V. (BITKOM). 2010. – Forschungsbericht