

Seminararbeit

Titel der Arbeit

Max Musterle

Semester: WS 20XX/20YY
Abgabedatum: TT. Monat 20YY
Betreuer: Prof. Dr. Alexander Wolff
Dr. Be TreuerInZwei
Be TreuerInDrei, M. Sc.



Julius-Maximilians-Universität Würzburg
Lehrstuhl für Informatik I
Algorithmen und Komplexität

1 Einleitung

Seminararbeiten und Praktikumsberichte an unserem Lehrstuhl sollen in dieser L^AT_EX-Vorlage erstellt werden. Dazu stellt die DocumentClass (erste Zeile) Parameter für die verschiedenen Arbeitsformen (`seminar`, `practical`) zur Verfügung. Für Abschlussarbeiten sollte stattdessen die Vorlage (`my-bachelor-master-zula.tex`) verwendet werden.

The localization of PDF bookmarks and such can be changed documentwide using the `english` parameter.

1.1 Git

Falls Sie zu zweit an Ihrer Seminausarbeitung arbeiten, lohnt es sich, auf dem Informatik-GitLab ein Repository mit dem Typ der Arbeit und Ihren Namen (also zum Beispiel „seminarausarbeitung-mueller-schmidt“) anzulegen. Dieser Vorlage liegt eine Datei namens `.gitignore` bei, welche `git` davon abhält, L^AT_EX-Hilfsdateien mit ins Repository zu schieben. Sollten Sie weitere Dateien im Repository-Ordner ablegen, welche nicht für alle gedacht sind und daher nicht hochgeladen werden sollten, können Sie die Dateinamen oder Ordner in `.gitignore` aufnehmen. Zum Beispiel sollte der Ordner mit den Leitfäden nicht mit ins Repository geladen werden – genauso wenig wie die PDF-Datei von Ihrer Ausarbeitung, die Sie mithilfe von L^AT_EX erzeugen.

1.2 Umgebungen

Wer bei uns eine Arbeit schreibt, kommt um Definitionen, Sätze, Beweise, Abbildungen und Tabellen nicht herum.

Definition 1. Dies ist eine *Definition*, welche sich sehr leicht erstellen lässt.

Eine kurze Überleitung von einer Definition zu einem Satz hilft dem Leser zu verstehen, wohin die Reise gehen soll.

Manchmal braucht man auch noch ein Lemma, damit der Beweis dann leichter von der Hand geht.

Lemma 2 (Andres Lemma). *Ein Lemma vorneweg erleichtert die Beweisführung.*

Beweis. Nach dem Distributivgesetz gilt:

$$(a + d) \cdot c = ac + dc$$

□

Satz 3 (Finkscher Hauptsatz). *Wichtige und grundlegende Sätze lassen sich leicht hervorheben.*

Beweis. Der Satz gilt offensichtlich, denn nach Lemma 2 gilt:

$$\sum_{i=1}^n 1 = n$$

Zudem wird der Beweis automatisch mit einem q.e.d.-Symbol¹ beendet. □

Auch wenn wir Lemma 2 und Satz 3 einen Namen verpasst haben, macht man das in der Regel aber eher nicht.

Korollar 4. *Aus einem Satz lassen sich oft einfache Resultate ableiten, die dann in einem Korollar festgehalten werden.*

1.3 Verweise

Auf Sätze, wie z.B. Satz 3, lässt sich mithilfe des Befehls `\cref{labelname}` verweisen, wenn man in der Satz-Umgebung ein „Label“ mit `\label{labelname}` gesetzt hat. Genauso können wir auf den nächsten Abschnitt, also Abschnitt 1.4, verweisen. Ist ein Verweis besonders weit weg, kann man `\cpageref{labelname}` verwenden, um dem Leser zu sagen, auf welche Seite er springen muss – Abschnitt 1.4 ist auf Seite 3. Wir benutzen hier das Paket (`cleveref`), welches nicht nur die Nummer sondern auch automatisch das richtige Label hinzufügt. Benötigt man nur die Nummer, so verwendet man `\ref{labelname}`. Üblicherweise beginnt man einen Labelnamen mit dem Typ der Umgebung, auf die man verweist, also z.B. `\label{fig:trapez}` für eine Abbildung (engl. *figure*). Ach ja, zum Hervorheben (engl. *emphasize*) eines *neuen Begriffs* verwendet man den Befehl `\demonstr{neuer Begriff}`, wenn der neue Begriff zum ersten Mal verwendet wird.

1.4 Gleitelemente

Gleitelemente (oder auch *floats*) *gleiten* durch das Dokument, und $\text{\LaTeX}$ ermittelt selbstständig eine geeignete Stelle im PDF, um diese anzuzeigen. Die wichtigsten Gleitelemente sind Abbildungen (`figure`), Algorithmen (`algorithm`) und Tabellen (`table`).

In diesem Abschnitt erläutern wir die einzelnen Gleitelemente. Eine Übersicht über die verschiedenen Gleitregeln findet sich am Ende in Tabelle 1.

1.4.1 Abbildungen

Abbildungen, wie z.B. Abbildung 1, sind schnell eingefügt. *Wichtig:* Fügen Sie eine Abbildung immer erst nach der ersten Referenz auf die Abbildung ein!

Im Allgemeinen braucht man die Endung der Bilddatei beim Einbinden mit dem Befehl `\includegraphics` nicht mit anzugeben. Es empfiehlt sich, alle Bilddateien in einem

¹Nach Möglichkeit sollte vermieden werden, dass der Beweis mit einer Leerzeile abgeschlossen wird, so wie es beim Beweis von Lemma 2 der Fall ist. Das kann zum Beispiel durch Hinzufügen einer Floskel wie „Damit ist die Aussage des Satzes bewiesen.“ erreicht werden. Übrigens funktionieren Fußnoten genau so. Wichtig ist hier, dass direkt vor dem Kommando kein Leerzeichen oder eine Leerzeile ist.

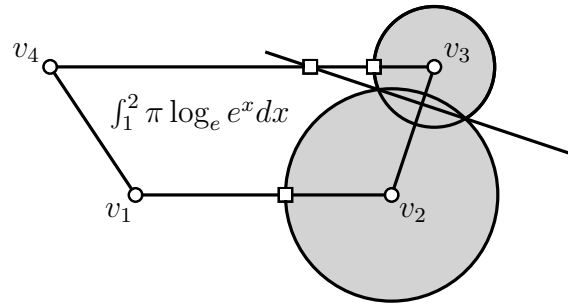


Abb. 1: Das ist eine Abbildung. Sie floatet nach der Regel *here* (h), d.h. L^AT_EX wird versuchen die Abbildung möglichst direkt vor dem folgenden Absatz („Im Allgemeinen...“) zu platzieren. Weitere Positionierungsregeln werden in Tabelle 1 erläutert.

Unterordner (z.B. **figures/**) abzulegen. In mehrteiligen Abbildungen kann man mit dem Paket **subcaption** jede Teilabbildung mit einer eigenen Bildunterschrift versehen. Man kann sich dann im Text sowohl auf die Teilabbildungen (z.B. Abbildung 2a) als auch auf die Gesamtabbildung (z.B. Abbildung 2) beziehen.

Bei Abbildungen ist die Bildunterschrift üblicherweise unter der eigentlichen Grafik. Für alle Gleitelemente gilt, dass das `\label{...}` direkt nach der `\caption{...}` gesetzt wird, damit bei einem Klick auf die Referenz (`\cref{...}`) auch an die richtige Stelle gesprungen wird (und nicht etwa auf den Absatz darüber). *Wichtig:* Man sollte sich im Text auf jede Abbildung wenigstens einmal beziehen.

1.4.2 Algorithmen

Mit der `algorithm`-Umgebung (aus dem Paket `algorithm2e.sty`) ist es nicht schwer, Algorithmen in Pseudocode zu setzen, siehe Algorithmus 1.

Funktionsnamen definiert man mit `\SetKwFunction{BinSearch}{BinäreSuche}`. Danach erscheint der Name über `\BinSearch{...}` überall einheitlich: im Algorithmus selbst, in anderen Algorithmen (z.B. bei rekursiven Aufrufen) und im Fließtext, wie hier

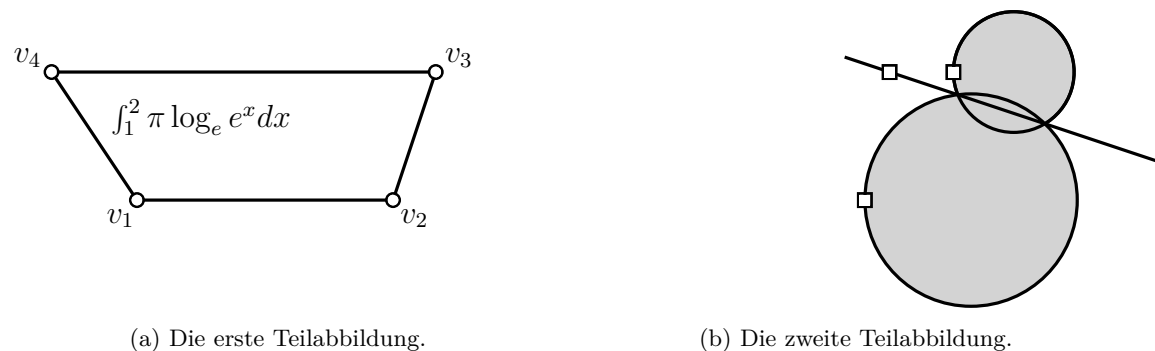


Abb. 2: Das ist eine Abbildung, die aus zwei Teilen besteht. Sie versucht erst *bottom* platziert zu werden, ansonsten erzeugt sie eine *float page*.

Algorithmus 1 : Binäre Suche in einem sortierten Feld

Eingabe : sortiertes Feld A , Länge n , gesuchtes Element x

Ausgabe : **true** genau dann, wenn x in A enthalten ist

```
1 BINÄRESUCHE( $A, n, x$ )
2    $l \leftarrow 0$ 
3    $r \leftarrow n - 1$ 
4   while  $l \leq r$  do
5        $m \leftarrow \lfloor (l + r)/2 \rfloor$ 
6       if  $A[m] = x$  then
7           return true
8       else if  $x < A[m]$  then
9            $r \leftarrow m - 1$ 
10      else
11           $l \leftarrow m + 1$ 
12  return false
```

bei `BINÄRESUCHE( $A, n, x$ )`.

Das gleiche geht problemlos auch ohne Zeilennummern, dazu benutzt man einfach in der `algorithm`-Umgebung den Befehl `\LinesNotNumbered`.

Für echten Quelltext (im Gegensatz zu Pseudocode) steht das Paket `listings` zur Verfügung: Code-Blöcke setzt man mit der Umgebung `lstlisting`, ganze Dateien bindet man mit `\lstinputlisting{datei}` ein und kurze Code-Stücke im Fließtext mit `\lstinline`.

1.4.3 Tabellen

Um $\text{\LaTeX}$ etwas in seiner Entscheidung zu beeinflussen, haben wir die Möglichkeit, für jedes Gleitelement einen oder mehrere Positionierungsparameter anzugeben. Eine Übersicht findet sich in Tabelle 1.

Innerhalb der eigentlichen „Tabelle“ befindet sich ein `tabularx`-Element, das den Inhalt darstellt. Elemente dieses Typs haben diverse Vorteile gegenüber normalen `tabular`-Elementen. Zum einen lässt sich die Breite der Tabelle festlegen (hier 90% der Textbreite), zum anderen stehen neben den klassischen zentrierten, links- oder rechtsbündigen (`l`, `r`, `c`) Spalten auch Spalten mit fester Breite zur Verfügung. Diese finden auch in Tabelle 1 Verwendung. Besonders zu beachten ist, dass sich Text in `p` oder `X` Spalten wie Text in normalen Text-Absätzen verhält, also mit automatischem Zeilenumbruch und Silbentrennung.

1.5 Literaturverweise

Auch Verweise auf andere Arbeiten, wie die von Mustermann und Musterfrau [MM11], sind ganz einfach. Wenn es von einem Artikel eine Konferenzversion [BDLN02] und eine

Tab. 1: Positionierungsparameter für Gleitelemente. Diese Tabelle selbst hat die Wahl zwischen *top* und *bottom*.

	Wert	Erklärung
[t]	<i>top</i>	Das Element wird oben auf einer Seite angezeigt.
[b]	<i>bottom</i>	Das Element wird unten auf einer Seite angezeigt.
[p]	<i>page</i>	Das Element wird auf einer neuen Seite angezeigt. Diese Seite wird auch <i>float page</i> genannt; sie kann mehrere [p]-Elemente beherbergen.
[h]	<i>here</i>	Das Element wird nah an der Stelle angezeigt, an der es im Code definiert ist. Das kann irgendwo auf der Seite sein.

Zeitschriftenversion [BDLN05] gibt, so sollten Sie stets die Zeitschriftenversion (oder ein Buch [GJ79]) zitieren. Tragen Sie bei einem Konferenzartikel neben Titel, Namen der Autoren und des Tagungsbandes gegebenenfalls auch die Namen der Herausgeber in die bib-Datei ein. ISBN-Nummern sind nicht nötig. Eine große Hilfe beim Identifizieren einer Quelle ist der Digital Object Identifier² – kurz *DOI* genannt. Sofern möglich, sollte stets der DOI einer zitierten Arbeit angegeben werden. Seien Sie vorsichtig mit Umlauten [BKNS10] und Unterstrichen in der bib-Datei.

Wichtig: Zitieren Sie so, dass der Satz, in dem ein Zitat vorkommt, auch ohne das Zitat noch Sinn ergibt; das Zitat soll eine Zusatzinformation sein.

Gut: Binucci et al. [BDLN05] beschäftigen sich mit der Beschriftung von Graphen.

Schlecht: [BDLN05] beschäftigt sich mit der Beschriftung von Graphen.

Möchte man einen klickbaren Link auf eine Website, wie zum Beispiel OpenStreetMap oder <http://www.graphdrawing.org/> setzen, verwendet man einfach `\href{link}{name}`.

1.6 Geladene Pakete

Die Klasse `info1thesis` lädt bereits die folgenden Pakete; diese müssen Sie nicht selbst laden. Wenn Sie Hilfe zu einem Befehl suchen, lohnt es sich, gezielt nach dem jeweiligen Paketnamen zu suchen.

Sprache und Schrift: `babel` (Silbentrennung, Bezeichnungen wie „Inhaltsverzeichnis“), `fontenc` und `lmodern` (Schriftart)

Mathematik und Sätze: `amsmath` (Formelumgebungen wie `align`), `amssymb` (zusätzliche Symbole), `amsthm` und `thmtools` (Satz- und Beweisumgebungen)

Verweise und Links: `hyperref` (klickbare Links, `\href`), `cleveref` (Verweise mit automatischem Namen, `\cref`)

²<https://www.doi.org/>

Abbildungen und Tabellen: `graphicx` (`\includegraphics`), `xcolor` (Farben), `caption` und `subcaption` (Bildunterschriften, Teilabbildungen), `booktabs` (Tabellenlinien wie `\toprule`), `tabularx` (Tabellen mit fester Breite)

Algorithmen und Code: `algorithm2e` (Pseudocode), `listings` (Quelltext)

Sonstiges: `enumerate` (angepasste Nummerierung von Aufzählungen), `subfiles` (Aufteilen der Arbeit in mehrere Dateien), `babelbib` (Literaturverzeichnis, in angepasster Form als `mybabelbib`)

Die genauen Optionen, mit denen die Pakete geladen werden, finden Sie in der Datei `info1thesis.cls`. Um ein einheitliches Aussehen der Arbeiten zu gewährleisten, sollte an diesen Einstellungen nichts geändert werden. Ist eine Anpassung in Absprache mit Ihren Betreuern doch einmal nötig, verwenden Sie die Setup-Befehle der Pakete (z.B. `\hypersetup` oder `\captionsetup`), statt das Paket mit `\usepackage` erneut zu laden – letzteres führt zu einem Fehler der Art *Option clash*.

Nicht automatisch geladen, aber in der Präambel dieser Vorlage bereits eingebunden, ist das Paket `todonotes`. Damit können Sie während des Schreibens ToDos direkt im PDF markieren: `\todo{...}` setzt eine Notiz an den Rand, `\todo[inline]{...}` einen ToDo-Block in den Text. Mit `\usepackage[disable]{todonotes}` blenden Sie alle ToDos aus, z.B. für die Abgabeversion (in der natürlich längst alle ToDos erledigt sind).

2 Fallstricke

Einige Sachen sollte man im L^AT_EX-Code vermeiden, um ein gut gesetztes Dokument zu erhalten. Schauen Sie sich dafür bitte auch die Schreibkonventionen an, welche im Ordner mit den Leitfäden (`guides/`) zu finden sind.

In Tabelle 2 sehen wir eine klassische *badbox*, also ein Element, das zu klein für seinen Inhalt ist. Mit einer Spalte vom Typ `p` oder `X` wäre das nicht passiert.

Tab. 2: Der Inhalt einer Tabelle sollte mit Bedacht gewählt werden.

Hier ein kleines Beispiel: Dieser Text wird sehr lang, wahrscheinlich sogar zu lang für eine Zeile. So kön

Auch unschön sind Zeilenumbrüche in Formeln, die sich im Fließtext befinden. $G = (V, E)$ ein Graph ist ein klassisches Beispiel dafür. Zu allem Überfluss fängt der vorherige Satz auch noch mit Formelsatz an – besser wäre „Sei $G = \dots$ “ gewesen. Möchte man verhindern, dass Formeln umgebrochen werden, setzt man ihren Inhalt in geschweifte Klammern.

Packt man den Inhalt in geschweifte Klammern, bleibt alles in einer Zeile. Sei $G = (V, E)$ ein Graph ist ein klassisches Beispiel dafür. Achtet man nun aber genau auf den rechten Rand dieses Absatzes, so sieht man, dass die Formel über den Rand des Blocksatzes hinausragt – wir können also das selbe Problem wie oben haben.

In beiden Fällen hilft oft ein einfaches Umstellen der Satzbausteine oder ein kurzes Füllwort.

Setzt man eine Fußnote nach einem Whitespace³, erhält man einen unschönen Abstand zwischen der Markierung und dem Wort, auf das sich die Fußnote wahrscheinlich bezieht.

Punkte sind nicht immer Satzenden, und das sollte man L^AT_EX auch mitteilen! Es gibt einen typographischen Unterschied zwischen `Prof. Wolff` und `Prof.\ Wolff`, der besonders beim Reservieren von Platz sichtbar wird.

Anführungszeichen setzt man in deutschen Texten mit "‘ und "’ („so“), in englischen Texten mit ‘‘ und ’’ (“so”). Das Zeichen " der Tastatur allein ergibt keine korrekten Anführungszeichen.

Möchte man verhindern, dass Wort und Referenz-Link in unterschiedlichen Zeilen landen, sollte man statt eines normalen Leerzeichens eine Tilde (~) verwenden.

3 Hinweise für Seminararbeiten

Eine Seminausarbeitung sollte auf keinen Fall nur eine „Übersetzung“ der Originalarbeit sein, sondern ihr gegenüber einen Mehrwert aufweisen. Hier ein paar Denkanstöße:

- Versuchen Sie die Darstellung zu vereinfachen und insbesondere Beweise verständlicher zu gestalten. Wenn man einen Beweis zum ersten Mal liest, gibt es oft Stellen, die man nicht auf Anhieb versteht. Schildern Sie solche Stellen ausführlicher als in der Originalarbeit! Oft ist es auch hilfreich, eine zusätzliche Illustration hinzuzufügen, den Ablauf eines Algorithmus anhand eines Beispiels zu erläutern usw.
- Es geht *nicht* darum die *gesamte* Originalarbeit ausführlicher darzustellen – Sie sollen sich ja schließlich auch an ein Seitenlimit halten. Es ist Teil Ihrer Aufgabe die wichtigsten Ideen der Originalarbeit zu identifizieren und eben jene sauber und ausführlich darzustellen. Eher nebensächliche Aspekte können abgekürzt oder sogar ausgelassen werden. Zum Beispiel können Beweise mancher Lemmata ausgelassen oder nur skizziert werden (`\begin{proof}[Beweisskizze] ... \end{proof}`). Besprechen Sie Ihre Ideen mit Ihren Betreuern!
- Diskutieren Sie Fragen, die in der Originalarbeit offen gelassen wurden! Warum funktioniert der angegebene Algorithmus nur für positive Kantengewichte? Was müsste man beachten, wenn auch negative Kantengewichte zugelassen sind?
- Oft haben die Artikel, die im Rahmen eines Seminars besprochen werden, ein gemeinsames Oberthema. Stellen Sie in Ihrer Arbeit Querbezüge zu den verwandten Artikeln her!
- Führen Sie eine eigene Literaturrecherche durch! In der Einleitung und dem Schlussteil Ihrer Ausarbeitung sollten Sie nicht nur auf Artikel verweisen, die auch in der

³Whitespaces sind insbesondere Zeilenumbrüche und Leerzeichen.

Originalarbeit referenziert werden. Prüfen Sie insbesondere, ob die vorliegende Arbeit bereits von anderen Autoren zitiert wurde und in welchem Kontext. Dazu kann man z.B. die Webseite Google Scholar verwenden (dort nach dem Artikel suchen und dann auf „Zitiert von:“ klicken).

Literatur

- [BDLN02] Carla Binucci, Walter Didimo, Giuseppe Liotta und Maddalena Nonato: Labeling Heuristics for Orthogonal Drawings. In: Petra Mutzel, Michael Jünger und Sebastian Leipert (Herausgeber): *Proc. 9th Int. Symp. Graph Drawing (GD'01)*, Band 2265 der Reihe *Lect. Notes Comput. Sci.*, Seiten 139–153. Springer-Verlag, 2002, 10.1007/3-540-45848-4_12.
- [BDLN05] Carla Binucci, Walter Didimo, Giuseppe Liotta und Maddalena Nonato: Orthogonal Drawings of Graphs with Vertex and Edge Labels. *Computational Geometry: Theory & Applications*, 32(2):71–114, 2005, 10.1016/j.comgeo.2005.02.001.
- [BKNS10] Michael A. Bekos, Michael Kaufmann, Martin Nöllenburg und Antonios Symvonis: Boundary Labeling with Octilinear Leaders. *Algorithmica*, 57(3):436–461, 2010, 10.1007/s00453-009-9283-6.
- [GJ79] Michael R. Garey und David S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York, 1979.
- [MM11] Max Mustermann und Monika Musterfrau: Beispiele in der Anwendung. *Beispiele und Muster*, 61(2):306–320, 2011.