

various

Lecture Notes of Spring 2013

Algorithms I

University of Mannheim
2013

This script originates from the course "Algorithms I" at the University of Mannheim as lecture notes.

The accuracy of its content is not guaranteed and the author(s) do not assume responsibility for possible damages of any kind. This lecture notes are not an official document released by employees of the University of Mannheim, hence those do not assume responsibility, as well.

Many thanks to Ingo Bürk, who initially published the underlying LaTeX template [here](#). This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License](#).

Contents

1	Benutzung der Skriptvorlage	5
1.1	Installation	5
1.2	Einstellungen zu Rechtshinweisen, Vorwort und Titelseite	6
1.3	Einstellungen	7
1.4	Verfügbare Makros	8
1.5	Verfügbare Umgebungen	10
2	Zum Schreiben eines Buches	13
2.1	Allgemeines	13
2.1.1	Modularer Aufbau	13
2.1.2	Stichwortverzeichnis	14
2.1.3	Trennungsregeln	14
2.1.4	Mathematischer Inhalt	14
2.1.5	Nützliche Links	16

1

Benutzung der Skriptvorlage

┌

In diesem Kapitel wollen wir die vielfältigen Möglichkeiten, die von dieser Vorlage bereitgestellt werden, kennenlernen und uns so einen Eindruck über ihre Möglichkeiten verschaffen. Wir werden sehen, wie diese Vorlage korrekt eingestellt und gehandhabt werden kann, um schöne Ergebnisse zu erzielen.

└

1.1 Installation

Da man die Vorlage gegebenenfalls mehrfach verwenden möchte und "Skriptvorlage" ein eher weniger aussagekräftiger Projekttitel ist, sollte darauf verzichtet werden, direkt in die Vorlagedateien zu schreiben.

Stattdessen sollte zunächst ein neues Projekt angelegt werden¹. In die Hauptdatei wird dann per "Copy & Paste" der Inhalt der Skriptvorlage.tex kopiert. Auf diese Weise legt man sich eine neue Kopie der Vorlage an und kann sie gleich entsprechend benennen. Die übrigen Dateien können dann einfach in das neue Verzeichnis kopiert werden. Nach diesen einfachen Schritten ist die Vorlage gewissermaßen "installiert" und als neues Projekt angelegt worden.

Der Inhalt sollte nun kapitelweise (siehe 2.1.1) in Teildokumente aufgeteilt werden, die im Ordner content/ untergebracht werden können. In der Hauptdatei muss dann an der entsprechenden, gekennzeichneten Stellen für jede Datei ein \include-Befehl gesetzt werden, um die Datei einzubinden. Es gilt zu beachten, dass beim Einbinden einer neuen Datei stets eine neue (rechte) Seite begonnen wird².

¹ Falls die verwendete Editor-Umgebung dieses Feature zur Verfügung stellt.

² Da der Befehl \cleardoublepage aufgerufen wird, werden insbesondere auch alle noch nicht ausgegebenen Figuren ausgegeben.

1.2 Einstellungen zu Rechtshinweisen, Vorwort und Titelseite

Die Dateien des sogenannten frontmatter-Bereiches³ im Ordner `inc/` sollten an die eigenen Ansprüche angepasst werden. Dies umfasst die folgenden Dateien:

- i) `copyright.tex` – für die Lizenzbestimmungen. Als Standard ist eine "Creative Commons"-Lizenz eingestellt.
- ii) `preface.tex` – für das Vorwort des Buches. Hier kann der Inhalt des Buches zusammengefasst, auf das Thema hingeleitet oder das Thema motiviert werden. Das Vorwort sollte keinesfalls aus mehreren Unterkapiteln bestehen oder andersweitig Merkmale aufweisen, die den Charakter des Inhaltes besitzen.
- iii) `titlepage.tex` – für die Titelseite. In der Regel sollte diese Datei nicht verändert werden, es sei denn, es ist explizit ein anderes Titelseitenlayout erwünscht.

³ Falls dieser über den `\usrmatter`-Schalter (siehe 1.3, vii)) überhaupt verwendet wird. Andernfalls existiert er zwar dennoch, erhält aber keine separate Nummerierung.

1.3 Einstellungen

In der Datei `inc/settings/settings.tex` finden sich alle wichtigen Einstellungen, die von dieser Vorlage angeboten werden, um das Layout oder Verhalten zu bestimmen. Ungeübten \LaTeX -Anwendern wird empfohlen, nur diese Einstellungen zu variieren und für Änderungen innerhalb der Vorlage Hilfe zu konsultieren, um unerwünschte Effekte zu vermeiden. Manche Schalter nehmen lediglich die Werte 0 und 1 an⁴. Der Wert 1 aktiviert, der Wert 0 deaktiviert den entsprechenden Schalter.

An dieser Stelle wollen wir im Detail auf die einzelnen Schalter und Optionen in der eben erwähnten Datei eingehen:

- i) `\docAutor`, `\docTitel`, `\docUntertitel`, `\docDozent`, `\docJahr`, `\docUniversitaet` – mit diesen Befehlen wird die Ausgabe auf der Titelseite⁵ gesteuert und automatisch angepasst.
- ii) `\docTitelZitat`, `\docTitelZitatName` – diese beiden Befehle betreffen ebenfalls die Titelseite. Es kann ein Zitat angegeben werden, das standardmäßig unter dem Titel erscheint. Aufgrund der unterschiedlichen Formatierung wird der Name der zitierten Person getrennt gespeichert. Der Befehl darf jedoch auf keinen Fall gelöscht werden, sondern muss einfach mit einem leeren Inhalt definiert werden.
- iii) `\useIndex` – mit diesem Schalter kann festgelegt werden, ob ein Stichwortverzeichnis ausgegeben werden soll.
- iv) `\useThumbs` – dieser Schalter aktiviert kleine Kästchen am rechten Seitenrand, die bei gedruckter Form helfen, bequem das richtige Kapitel aufzuschlagen. Es gilt jedoch zu beachten, dass nur wenige Drucker den Rand wirklich bedrucken können.
- v) `\useRoman` – Mit dieser Einstellung werden Kapitel, die mit `\chapter` eröffnet werden, mit römischen statt arabischen Ziffern nummeriert.
- vi) `\usrBCOR` – da beim Binden eines Buches Teile des inneren Randes augenscheinlich verschwinden wird die Bindekorrektur (BCOR, engl. *binding correction*) eingeführt.
- vii) `\usrmatter` – ist diese Option aktiviert, so werden die Seiten vor dem eigentlichen Inhalt des Buches separat nummeriert und in römischen Zahlen ausgegeben. Die Nummerierung beginnt ab dem eigentlichen Inhalt dann neu.
- viii) `\usroptsqrt` – mit diesem standardmäßig aktiviertem Schalter wird die Form des Wurzelsymbols angepasst, so dass ein schließender Strich eingefügt wird: \sqrt{x} versus \sqrt{x}
- ix) `\usrnscmd` – hier wird der Befehl eingestellt, der für die Zahlenraum-Makros `\A`, `\dots`, `\Z` verwendet wird (siehe 1.4). Eine gängige Alternative ist `\mathbb`, schöner wäre jedoch `\mathds`. Als Standard wird eine simple Fettschrift gesetzt, da dies auch der traditionellen Form entspricht.

⁴ Tatsächlich kann man ihnen beliebige Werte zuweisen, lediglich der Wert 1 aktiviert den Schalter jedoch.

⁵ In der Standarddatei für die Rechtshinweise werden diese Befehle ebenfalls verwendet.

1.4 Verfügbare Makros

Diese Vorlage bietet standardmäßig bereits viele vordefinierte Makros und Umgebungen. Die Definitionen der Makros werden in der Datei `inc/settings/abbreviations.tex` verwaltet. Je nach persönlichen Präferenzen und der Notwendigkeit kann diese vordefinierte Liste verändert oder erweitert werden.



Makros sind ein wichtiges Werkzeug, um einen konsistenten Stil zu gewährleisten. Sie erleichtern zudem die Arbeit, da etwaige Änderungen nur einmalig und nicht mühsam im gesamten Dokument durchgeführt werden müssen.

An dieser Stelle wollen wir alle verfügbaren Makros kennenlernen:

- i) Die Befehle `\A, \dots, \Z` liefern Buchstaben für Zahlenräume, lediglich `\H`, `\L`, `\O` und `\P` stehen nicht standardmäßig zur Verfügung. Der Befehl `\I` ist ebenfalls andersweitig belegt (siehe iv)). Es existieren nur Großbuchstaben mit diesen Befehlen. Die Makros verwenden intern den Befehl `\usrnscmd`⁶, der standardmäßig auf `\textbf` eingestellt ist.

A, B, C, D, E, F, G, J, K, M, N, Q, R, S, T, U, V, W, X, Y, Z.

Weit verbreitet sind auch die Befehle `\mathbb` und `\mathds` für doppelgestrichene Buchstaben. Historisch gesehen wurden diese aber nur eingeführt, da Fettschrift an der Tafel schwer umzusetzen war, weswegen diese Varianten eigentlich nicht verwendet werden sollten. Da sie dennoch sehr verbreitet sind werden sie unterstützt. Es wird empfohlen, in diesem Fall `\mathds` zu verwenden, da die Doppelstriche an der korrekten Position sind und die Schriftzeichen die richtige Größe haben.

- ii) Analog liefert `\sA, \dots, \sZ` kalligrafisch geschwungene Buchstaben. Auch hier gibt es nur Großbuchstaben, es existieren hierbei aber sogar alle Buchstaben. Intern verwendet wird der Befehl `\mathcal`.

A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z.

- iii) Mit dem Befehl `\deftxt{\ldots}` können definierte Begriffe hervorgehoben werden. Wir nennen dies einen `\deftxt{Begriff}` liefert:

Wir nennen dies einen **Begriff**

- iv) Für die imaginäre Einheit i existiert das Makro `\I`.
v) Für das Differential d in Integralen steht das Makro `\dd` zur Verfügung.
vi) Für das häufig verwendete ε gibt es den Befehl `\e`.
vii) Mit `\id` erhält man den Identitätsoperator id .

⁶ Siehe 1.3, ix).

- viii) Für die Indikatorfunktion kann `\ind_A(x)` verwendet werden und liefert dann $\mathbf{1}_A(x)$.
- ix) Eine Norm kann mit `\norm{x}` geschrieben werden und liefert $\|x\|$. Die Normstriche passen sich in der Größe automatisch an.
- x) Der Realteil $\operatorname{Re} z$ und der Imaginärteil $\operatorname{Im} z$ komplexer Zahlen lassen sich mit `\Re z` und `\Im z` notieren, verhalten sich also genauso wie die gängigen Operatoren wie z. B. `\sin x` für $\sin x$.
- xi) Für Notationen der Stochastik existieren `\Var` für die Varianz Var , `\Cov` für die Kovarianz Cov und `\Pois` für die Poisson-Verteilung Pois .

1.5 Verfügbare Umgebungen

Nachdem wir nun die Makros kennen, die von der Vorlage bereitgestellt werden, kommen wir zu den verfügbaren Umgebungen. Ihre konsequente Verwendung wird für einen typographisch guten Stil dringend empfohlen:

- i) Definitionen können mit der definition-Umgebung erstellt werden.
`\begin{definition} \ldots \end{definition}` liefert:

Definition 1.5.1

...

Die Nummerierung erfolgt automatisch entsprechend des aktuellen Kapitels. Alle Umgebungen teilen sich einen "Zähler", so dass jede Nummer einzigartig ist.

- ii) Für Sätze, Korollare, Lemmata und Propositionen gibt es entsprechende Umgebungen `vermöge satz`, `korollar`, `lemma` und `proposition`. Analog zu der `definition`-Umgebung liefert dies beispielsweise:

Satz 1.5.2 Satz von ...

...

- iii) Für Beispiele stehen die Umgebung `beispiel` bzw. `beispiel*` für unnummerierte Beispiele zur Verfügung:

Beispiel 1.5.3

... //

Endet ein Beispiel beispielsweise in einer Liste oder in einer abgesetzten Gleichung, so kann mit `\qedhere` der Endmarker manuell an die entsprechende Stelle gesetzt werden.

All diese Umgebungen erlauben zudem einen optionalen Parameter in eckigen Klammern – also z. B. `\begin{satz}[Bezeichnung]` –, welcher für den Titel der Umgebung gewählt wird. Je nach Umgebung kann dies eine Bezeichnung des Theorems oder der zu definierende Begriff sein. Ein Beispiel hierfür wurde eben schon gezeigt.



Die Umgebungen sichern sich mit Hilfe des `needspace`-Pakets bereits Platz, dennoch kann es unter Umständen zu leichten Darstellungsfehlern kommen, bei denen oben oder unten der Leerraum zum farbigen Hintergrund fehlt. Dieser Fehler lässt sich bisher leider nur durch redaktionelle Eingriffe im Einzelfall beheben.

- iv) Für Beweise steht die `beweis`-Umgebung zur Verfügung. Ihr Aussehen ist schlicht gehalten. Das `\qed`-Symbol wird automatisch eingefügt; sollte der Beweis jedoch mit einer

abgesetzten Gleichung enden, so sollte der Befehl `\qedhere` innerhalb dieser Gleichung verwendet werden, um das Symbol korrekt zu positionieren.

Proof: Hier steht ein Beweis. □

- v) Die Umgebung `information` dient für informative Hinweise neben des eigentlichen Themas. So können hiermit interessante Ausblicke in andere Anwendungen, Fachgebiete oder auch in die Geschichte zu einem Thema gegeben werden.



Diese Umgebung sollte gezielt eingesetzt werden, um den Lesestoff zwar aufzulockern, jedoch nicht vom eigentlichen Thema abzulenken. Der Inhalt sollte informativ, aber kurzgehalten sein.

- vi) Die `\beschreibung-Umgebung` sollte zu Beginn eines jeden Kapitels – also nach dem Aufruf des `\chapter`-Befehls – verwendet werden, um einen einheitlichen Stil im Buch zu erhalten (oder, andernfalls, gar nicht verwendet werden). Sie dient dazu, den Inhalt des kommenden Kapitels kurz zusammenzufassen und so einen Einblick in die bevorstehende Thematik zu gewähren. Ein Beispiel für diese Umgebung findet sich zu Beginn dieses Kapitels.

2

Zum Schreiben eines Buches

┌

Nachdem wir nun die spezifischen Grundlagen und Möglichkeiten der Vorlage kennen, wollen wir noch auf einige grundlegende Regeln eingehen, die man beim Verfassen eines Buches einhalten sollte. Ferner gehen wir darauf ein, wie ein großes Projekt richtig gehandhabt werden kann.

└

2.1 Allgemeines

2.1.1 Modularer Aufbau

Um die Übersicht über ein großes Projekt zu behalten ist es sinnvoll, dieses modular aufzubauen, also in Teildokumente zu splitten. Mit dieser Vorlage sollte dabei für jedes Kapitel (`\section{}`) ein eigenes Dokument angelegt werden, wie es z. B. in diesem Beispieldokument zu sehen ist, da der verwendete `\include`-Befehl jede Datei auf eine neue Seite setzt.

Neben typographischen Gründen erlaubt uns dies, ein für große Dokumente sehr nützliches \LaTeX -Feature – `\includeonly{}` – zu nutzen. Um das Dokument nicht jedesmal komplett kompilieren zu müssen kann man \LaTeX nur die benötigten Teile kompilieren lassen; Seitenzahlen, Referenzen etc. gehen dabei aber nicht verloren. In der Hauptdatei des Projektes findet sich dazu der (standardmäßig auskommentierte) Befehl `\includeonly{}`, in welchen man – durch Kommata getrennt – die Dateien angeben kann, die kompiliert werden sollen. In der finalen Version wird der Befehl dann einfach wieder auskommentiert.

Sollte es nicht erwünscht sein, neue Unterkapitel auf neue Seiten zu platzieren, so müssen die Unterdateien in der Hauptdatei per `\input` statt mit `\include` eingebunden werden. Jedoch kann das `\includeonly`-Feature dann nicht mehr genutzt werden, da es für `\input` kein entsprechendes Äquivalent dazu gibt.

2.1.2 Stichwortverzeichnis

Um in einem großen Dokument Begriffe schnell nachschlagen zu können ist ein Stichwortverzeichnis eine gute Möglichkeit. Diese Vorlage kreiert standardmäßig auch ein solches, die Einträge müssen jedoch manuell angelegt werden. Dazu verwendet man den Befehl `\index{}` unmittelbar und ohne Leerzeichen hinter dem jeweiligen Wort:

Dies ist ein Begriff`\index{Begriff}` und wir...

Mehr zu diesem Thema kann man der Dokumentation des Paketes `makeidx` entnehmen.



Die Indizierung für das Stichwortverzeichnis sollte nicht schon während des Schreibens durchgeführt werden. Stattdessen sollte erst das gesamte Buch geschrieben und dann indiziert werden, da es während des Schreibens schwierig ist, abzusehen, welche Begriffe wichtig genug sind, um sie im Verzeichnis aufzulisten.

Sollte kein Stichwortverzeichnis erwünscht sein, so muss `\useIndex` in den Einstellungen (also in der `inc/settings.tex`) auf 0 gesetzt werden.

2.1.3 Trennungsregeln

Ab und an kann es passieren, dass die Trennungsregeln, die \LaTeX mitliefert, versagen und ein Wort nicht korrekt umgebrochen wird (was z. B. in einer "overfull \hbox"-Warnung resultieren kann). Um diesen Effekt auszugleichen wird bereits das Paket `microtype` verwendet. Sollte auch dieses versagen, so sollte von manuellen Zeilenumbrüchen zunächst abgesehen werden. Stattdessen kann und sollte man besser eine Regel für dieses Wort in der Datei `inc/settings/hyphenation.tex` angeben. Dies geschieht z. B. für das Wort "Trennungsregel" mit dem folgenden Eintrag:

```
\hyphenation{Trenn-ungs-re-gel}
```

2.1.4 Mathematischer Inhalt

Wir wollen hier nicht im Detail darauf eingehen, wie man mathematische Formeln in \LaTeX realisiert. Es gibt jedoch einige prinzipielle Dinge, die wir ansprechen wollen:

- i) Für abgesetzte Formeln sollte primär die `align`-Umgebung (bzw. `align*` für unnummerierte Gleichungen) verwendet werden. Die Umgebung `eqnarray` und andere erzeugen inkonsistente Abstände und sollten gemieden werden¹.
- ii) Besonders lange Formeln, die nicht in eine Zeile passen, können mit der `multline`- bzw. `multline*`-Umgebung realisiert werden.

¹ Dieser und andere typische Fehler finden sich im \LaTeX -Sündenregister (siehe 2.1.5).

iii) Diese Vorlage bindet das Paket `mathtools` ein, welches einige zusätzliche Befehle bereitstellt. Erwähnenswert sind insbesondere:

- a) `\xrightarrow[]{}{}` für beschriftete Pfeile, die sich automatisch in der Länge anpassen. Der Befehl `\xrightarrow[x]{y}` liefert so z. B. die Ausgabe $\xrightarrow[x]{y}$. Nähere Informationen können der Dokumentation entnommen werden.
- b) Um mehrzeilige Formeln mit Text zu versehen, ohne dabei die Ausrichtung zu verlieren, können in \LaTeX standardmäßig mit dem Befehl `\intertext{}` vorübergehend unterbrochen werden. Da dieser Befehl jedoch sehr große Abstände erzeugt kann bei Bedarf der nun verfügbare Befehl `\shortintertext{}` verwendet werden, der kleinere vertikale Abstände erzeugt.
- c) Bei – zum Beispiel – Summen mit langen Indexausdrücken schafft `\mathclap` Abhilfe. Der Befehl sorgt dafür, dass das übergebene Argument für \LaTeX keine Breite einnimmt und lässt sich somit für schönere Positionierungen nutzen. Im folgenden Beispiel wurde der Befehl gleich zweimal verwendet:

$$a = z + y + x + \sum_{i \leq j \leq n+1} ij + \dots$$

$$\stackrel{\text{ABC}}{=} p.$$

Das selbe Beispiel ohne die Verwendung des Befehls sieht wie folgt aus:

$$a = z + y + x + \sum_{i \leq j \leq n+1} ij + \dots$$

$$\stackrel{\text{ABC}}{=} p.$$

In der zweiten Zeile beachte man die horizontale Position des Gleichheitszeichens im Vergleich zu dem aus der ersten Zeile.



Es lohnt sich in jedem Fall, die entsprechenden Paketdokumentationen zu lesen, um ihr jeweiliges Potenzial zu nutzen. Auch ein Blick in das bereits erwähnte \LaTeX -Sündenregister (siehe 2.1.5) ist empfehlenswert.

Um das Aussehen mathematischer Formeln zu demonstrieren wollen wir hier ohne Angabe der Voraussetzungen die [Formel von Cauchy](#) und die [Residuenformel](#) angeben:

$$\frac{1}{2\pi i} \oint_{\Gamma} \frac{f(z)}{z-a} dz = f(a) \quad \text{und} \quad \oint_{\Gamma} f(z) dz = 2\pi i \sum_{k=1}^N n(\Gamma, a_k) \text{Res}_{a_k} f.$$

Auch bei Formeln im Fließtext, wie zum Beispiel $a \in [0, 1)$, kann man gut sehen, wie sich der speziell zur Fließtextschriftart passende mathematische Satz in das Schriftbild einfügt.

2.1.5 Nützliche Links

In der folgenden Liste finden sich hilfreiche und nützliche Links zu Dokumentationen oder anderen Dokumenten, die es wert sind, gelesen zu werden:

- [l2tabu.pdf](#) – das L^AT_EX-Sündenregister, das weiter oben schon erwähnt wurde. Es listet die gängigsten Todsünden und obsoleten Pakete auf und nennt dabei auch jeweils Alternativen.
- [typokurz.pdf](#) – eine Zusammenfassung grundlegender typografischer Regeln für besonders anspruchsvolle Autoren. Die Typografie beschäftigt sich insbesondere auch mit der Lesbarkeit von Texten und ist daher ein sehr nützliches Werkzeug.
- [T_EXblog.net](#) – der T_EX-Blog von Stefan Kottwitz, der auch das Buch "L^AT_EX – Beginner's Guide" geschrieben hat. Hier findet man wertvolle Informationen und Anleitungen rund um T_EX und L^AT_EX.
- [KOMA-Script](#) – die KOMA-Script-Klassen von Markus Kohm, die auch von dieser Vorlage verwendet werden, sind ein sehr umfangreiches und mächtiges Werkzeug im Umgang mit L^AT_EX. Ein Studium ihrer Möglichkeiten ist insbesondere dann zu empfehlen, wenn mit diesen Klassen gearbeitet wird, um bestmögliche Ergebnisse zu erzielen.