

Praktische Informatik 1

Dokumente erstellen mit LaTeX

Thomas Röfer

Cyber-Physical Systems
Deutsches Forschungszentrum für
Künstliche Intelligenz

Multisensorische Interaktive Systeme
Fachbereich 3, Universität Bremen



TeX und LaTeX

- **TeX** ist ein Compiler, der einen Quelltext in ein typographisch professionell gesetztes Dokument übersetzt
- Er folgt dabei Regeln, die Verstöße gegen Layout-Regeln bewerten und sucht jeweils den besten Kompromiss
- **TeX** wurde 1979 von Donald E. Knuth entwickelt, um damit seine Bücher „The Art of Computer Programming“ zu schreiben
- **LaTeX** ist ein Makro-Paket von Leslie Lamport aus dem Jahr 1986, das die Erstellung strukturierter Dokumente auf Basis von **TeX** unterstützt

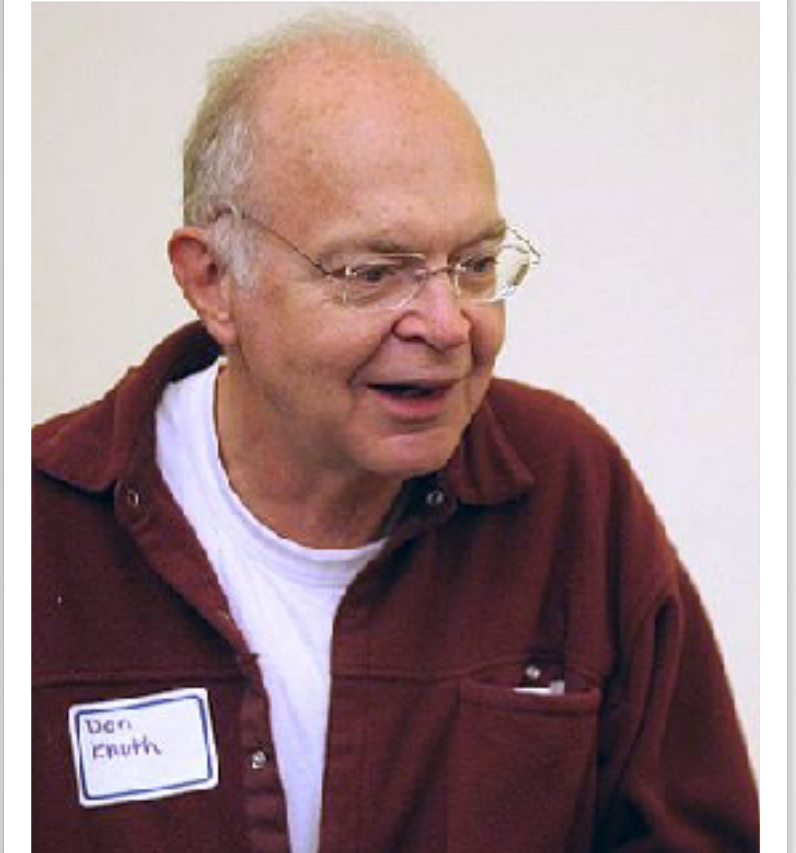
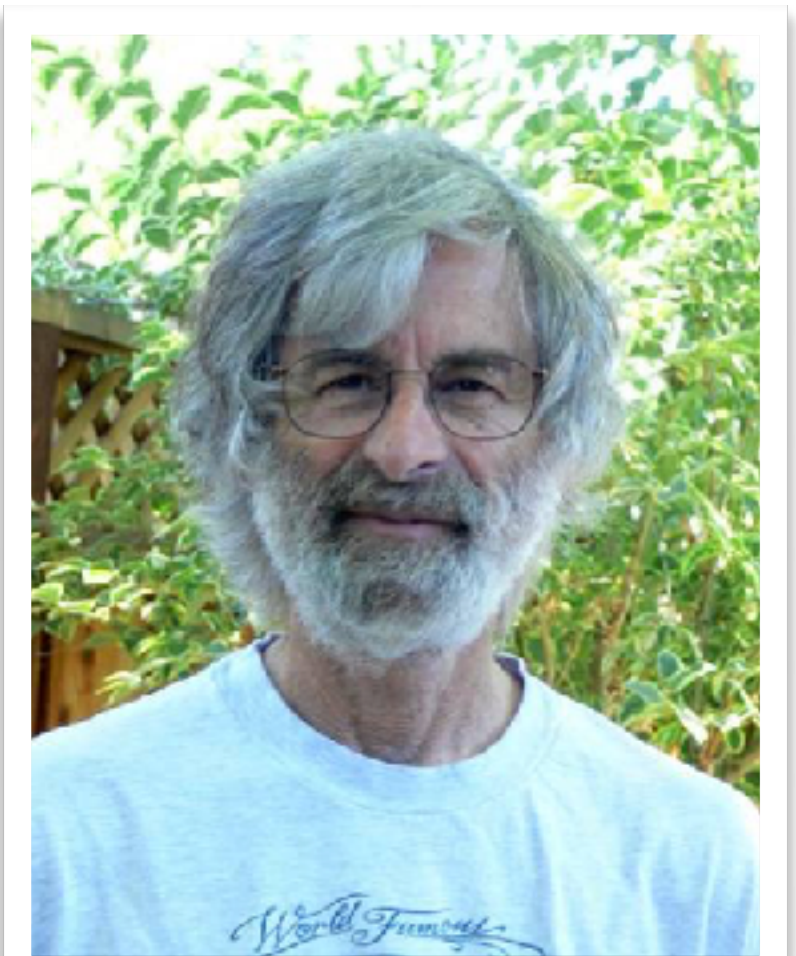
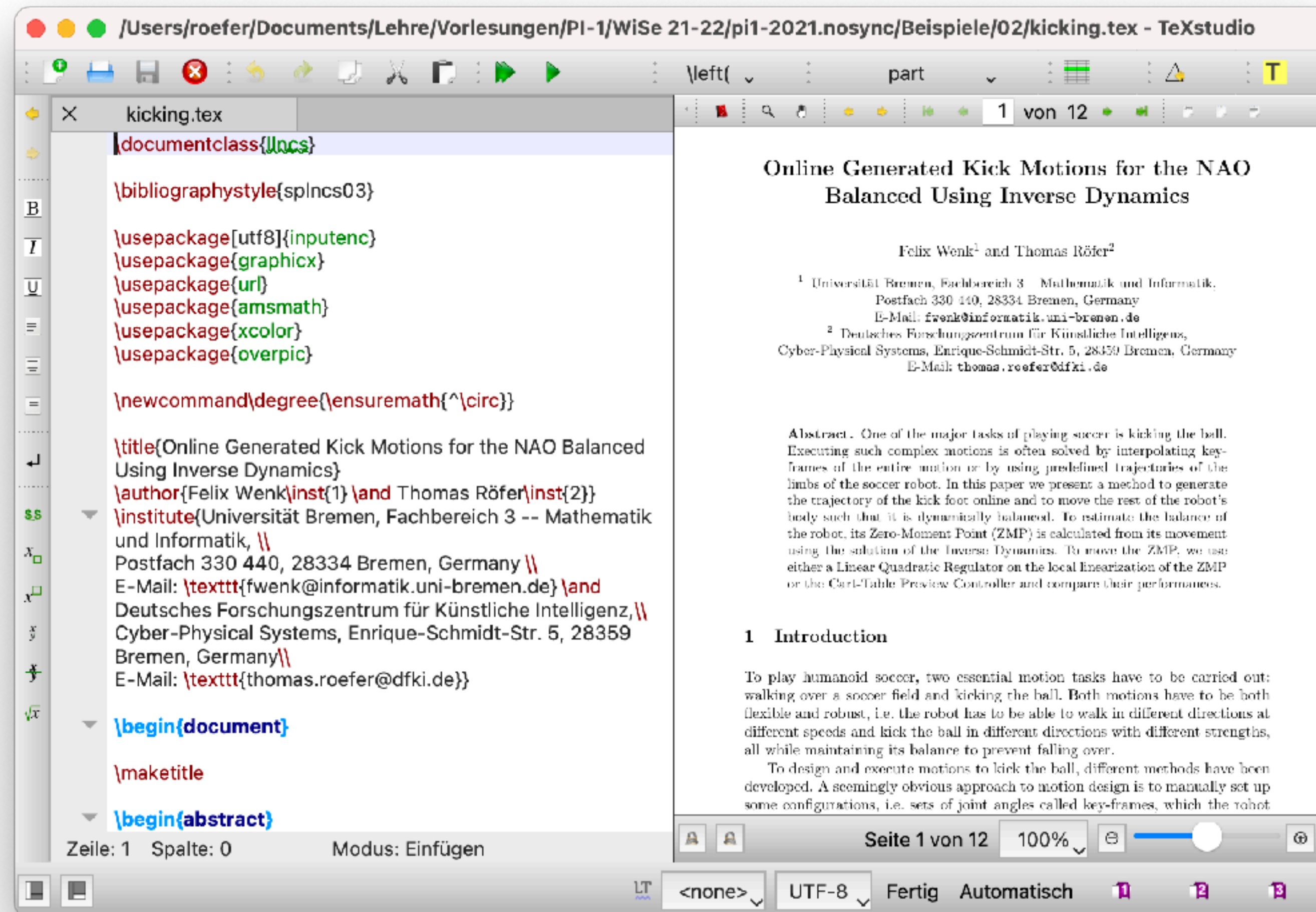


Foto aufgenommen von Jacob Appelbaum



Wissenschaftliches Papier: Demo





Reservierte Zeichen

- Bestimmte Zeichen braucht TeX, um die eigenen Strukturen zu erkennen. Als Text können sie meistens mit vorangestelltem **** eingegeben werden (z.B. **\%**)
- ****: Leitet Makros ein. Ersatz: **\textbackslash** oder **\backslash** (Mathematikmodus)
- **%**: Leitet Kommentare ein
- **{ ... }**: Gruppiert Text, z.B. für Makroparameter
- **\$**: Schaltet zwischen Normal- und Mathematikmodus um
- **&**: Trennt Zellen in tabellenartigen Strukturen
- **~**: Untrennbares Leerzeichen. Ersatz: **\textasciitilde** oder **\sim** (Mathematikmodus)
- Manche andere Zeichen können auch nicht direkt verwendet, aber durch ein Makro erzeugt werden

Befehle

- Beginnen mit ****, dahinter kommt ihr Name
- Können zwingende und optionale Parameter haben
 - Zwingende Parameter sind entweder nur ein Zeichen, ein LaTeX-Befehl oder ein Block **{ ... }**
 - Optionale Parameter werden in eckige Klammern **[...]** eingefasst
- Legale Werte für die Parameter sind pro Befehl individuell verschieden
 - Oft ist es auch einfach Freitext

```
\item  
\item[a]  
\frac48  
\frac{15}{16}  
\newcommand\xyz{XYZ}  
\begin{figure}[th]  
\includegraphics[width=12cm]{bild.pdf}
```

Präambel

- Die **Präambel** steht vor dem eigentlichen Text, der zwischen **`\begin{document}`** und **`\end{document}`** steht
- **`\documentclass{ ... }`** muss ganz am Anfang stehen und legt den Typ des Dokuments fest und damit seine Formatierung (z.B. **article**, **report**, **book**, bei uns **pi1**)
- **`\usepackage`** bindet Pakete ein, die bestehende Befehle ändern oder neue hinzufügen (kann mehrfach genutzt werden)
- Des Weiteren können Makros (um)definiert werden, Parameter der Vorlage gesetzt oder verändert werden uvm.

Nützliche Pakete

- **`\usepackage[utf8]{inputenc}`**: Mit UTF-8 kodierter Text wird verstanden (Umlaute, **ß** usw.)
- **`\usepackage[ngerman]{babel}`**: Deutsche Texte, deutsche Silbentrennung und typographische Anführungszeichen mit `"`` und `""`
 - Erzeugen von Umlauten mit `"a`, `"o`, `"u`, `"A`, `"O`, `"U` und **ß** mit `"s`
- **`\usepackage{graphicx}`**: Bilder einbinden
- Diese drei werden von der Dokumentenklasse **pi1** bereits eingebunden



Inhaltsstruktur

- Der Text wird in Abschnitte aufgeteilt, indem Überschriften definiert werden
 - **`\chapter{Kapitelüberschrift}`**: Beginnt neues Kapitel auf neuer Seite (**report**, **book**)
 - **`\section{Überschrift}`**, **`\subsection{...}`**, **`\subsubsection{...}`**: Abschnitte unterschiedlicher Hierarchie-Ebenen. **`\section*{...}`** usw. unterdrückt Nummerierung. **pi1** kennt noch **`\skipsection`**, um eine Aufgabe zu überspringen
 - **`\paragraph{Spitzmarke}`**, **`\subparagraph{...}`**
- Daraus kann mit **`\tableofcontents`** ein Inhaltsverzeichnis generiert werden
- Absätze werden durch eine **leere Zeile** beendet

Schriftauszeichnungen

- **\emph{...}**: Text so hervorheben, wie die Ersteller:in der Dokumentenklasse dies vorgesehen hat (meist *kursiv*)
- **\textbf{...}**: Fett (**boldface**)
- **\textit{...}**: Kursiv (*italics*)
- **\texttt{...}**: Feste Zeichenbreite (`typewriter`)
- Können gemischt werden, wenn der Schriftschnitt verfügbar ist (**\textbf{\textit{...}}**)

Titelseite bzw. -bereich

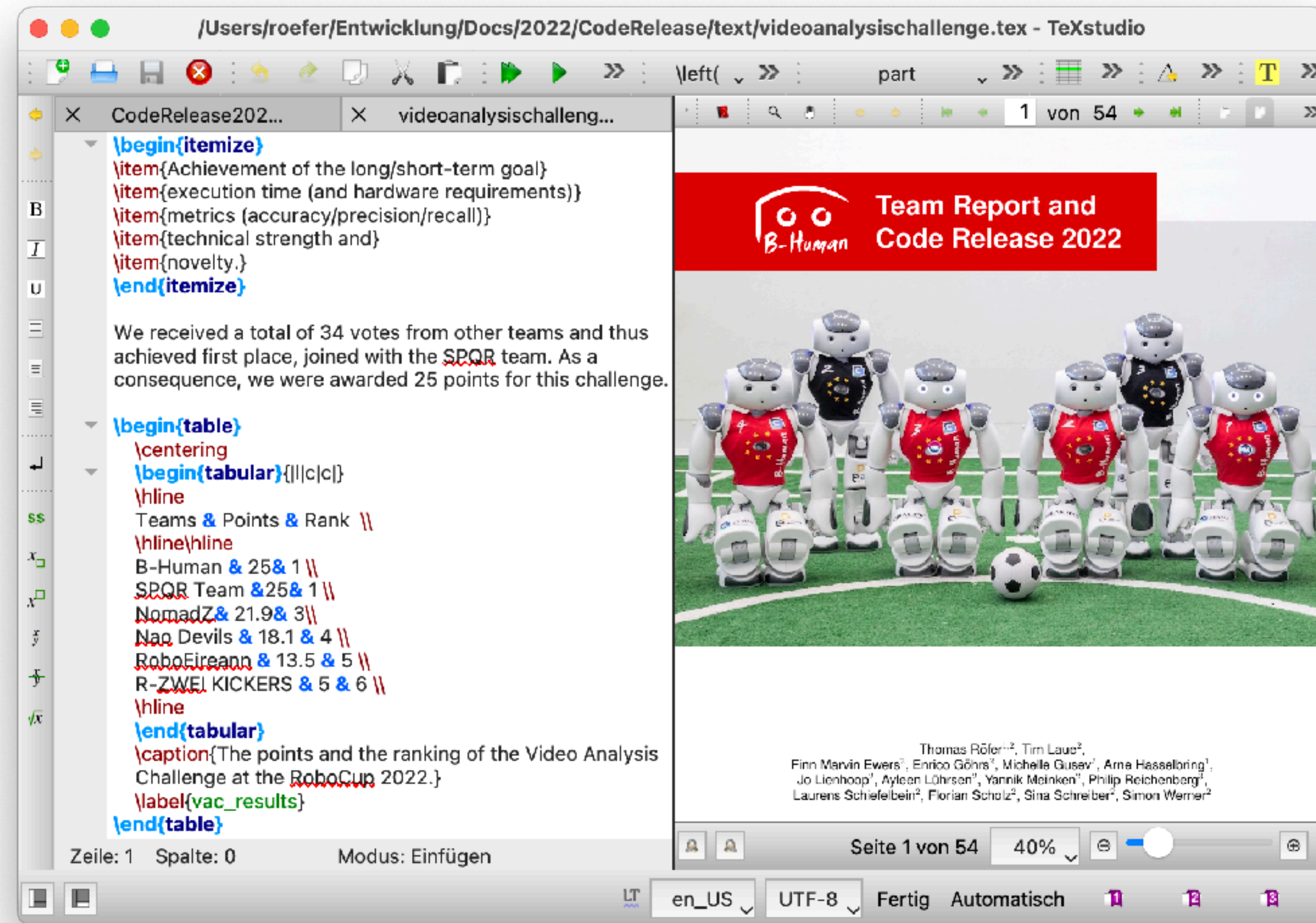
- Direkt auf **`\begin{document}`** folgt oft **`\maketitle`**
- Erzeugt bei **article** einen Titelbereich und bei **report** und **book** eine Titelseite
- Verwendet in der Präambel definierte **`\title{...}`**, **`\author{...}`** und **`\date{...}`**
- Achtung: Bei **pi1** werden die Parameter direkt übergeben:
`\maketitle{Übungsnummer}{Tutor:in}{Bearbeiter:in}`



Graphiken einbinden

- **`\usepackage{graphicx}`** in Präambel (ist in **pi1** enthalten)
- **`\includegraphics[Optionen]{Dateiname}`**
 - **`width`**=Breite, z.B. **`width=5cm`** oder **`width=0.8\textwidth`**
 - **`height`**=Höhe, z.B. **`height=4cm`** oder **`height=0.3\textheight`**
 - **`scale`**=Faktor, z.B. **`scale=0.5`**
 - **`angle`**=Winkel, z.B. **`angle=90`**
- Mehrere Optionen sind möglich und werden durch Kommas getrennt
- **pdfLaTeX** unterstützt die Grafikformate **.jpg** (Fotos), **.png** (z.B. Screenshots) und **.pdf** (Zeichnungen)

Umgebungen: Demo



Umgebungen

- **\begin{umgebung} ... \end{umgebung}** definiert einen Bereich im Dokument, in dem eigene Regeln gelten (anderes Layout, andere oder zusätzliche Befehle)
- **document, abstract**: Dokument und Zusammenfassung
- **itemize, enumerate, description**: Aufzählungen
- **tabular, tabbing**: Tabellen
- **figure, table**: Rahmen für „fließende“ Abbildungen und Tabellen
- **verbatim, listinglst**: Roher Text, Quelltext-Listing
- **equation, align**: Gleichungen

Umgebungen: Aufzählungen

- **`\begin{itemize}`** ... **`\end{itemize}`** definiert Bulletpoint-Liste
- **`\begin{enumerate}`** ... **`\end{enumerate}`** definiert nummerierte Aufzählung
- **`\begin{description}`** ... **`\end{description}`** definiert Liste aus Begriffen mit ihrer Definition (**`\item`**[Begriff] Definition)
- **`\item`** Definiert einen Aufzählungspunkt
- Können geschachtelt werden

```
\begin{itemize}  
\item A  
\item B  
\item C  
\end{itemize}
```

- A
- B
- C

```
\begin{enumerate}  
\item A  
\item B  
\item C  
\end{enumerate}
```

1. A
2. B
3. C

Tabellen

- **`\begin{tabular}`**{Spalten} ... **`\end{tabular}`**
erzeugt eine Tabelle
- Spalten können linksbündig (**`l`** oder **`p`**{feste breite}), rechtsbündig (**`r`**) oder zentriert (**`c`**) sein
- **`|`** in der Spaltendefinition fügt senkrechte Linien ein, **`\hline`** im Tabellenrumpf horizontale
- **`&`** trennt Zellen, **`\\`** trennt Zeilen

```
\begin{tabular}{|p{1cm}||rc|rc||}  
\hline  
Name & Pkt. & Note & Pkt. & Note & best. \\  
\hline  
\hline  
Peter-Paul & 70\% & 2,7 & 45\% & 5 & nein \\  
\hline  
Maria & 100\% & 1 & 100\% & 1 & ja \\  
\hline  
\end{tabular}
```

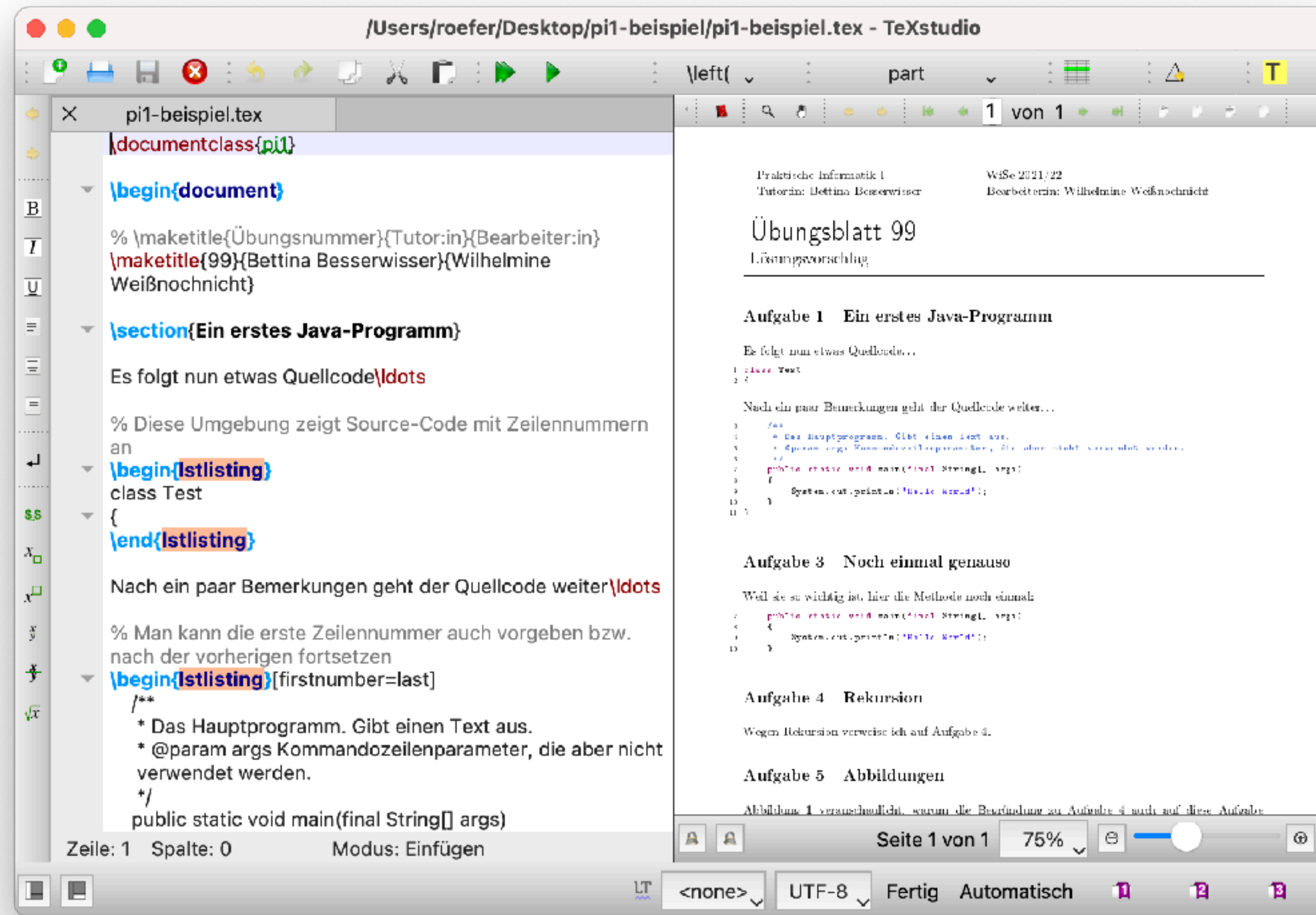
Name	Pkt.	Note	Pkt.	Note	best.
Peter-Paul	70%	2,7	45%	5	nein
Maria	100%	1	100%	1	ja

Fließende Abbildungen und Tabellen

- **`\begin{figure}[Platzierung] ... \end{figure}`**: Rahmen für Abbildungen, die LaTeX an eine passende Stelle platziert
- **`\begin{table}[Platzierung] ... \end{table}`**: Rahmen für Tabellen, die LaTeX an eine passende Stelle platziert
- Gewünschte Platzierung kann hier (**h**), am oberen (**t**) und unteren (**b**) Seitenrand bzw. auf einer eigenen Seite (**p**) sein, wobei mehrere angegeben werden können oder eine mit nachgestelltem **!** forciert werden kann
- **`\caption{...}`** definiert Beschriftung

```
\begin{figure}[ht]
\centering
\includegraphics[width=0.1\textwidth]{LaTeXLogo.png}
\caption{Das \LaTeX-Logo als Bild}
\end{figure}
```


PI-1-Vorlage: Demo



Quelltext-Listings

- **`\usepackage{listings}`** in Präambel (ist in **pi1** enthalten)
- **`\begin{lstlisting}[firstnumber=Anzeigezeile] ... \end{lstlisting}`**: Quelltexte einbinden, Zeilen durchnummerieren und Syntax hervorheben (Zeilenangabe ist optional, **`firstnumber=last`** setzt vorherige Zählung fort)
- **`\lstinputlisting[firstnumber=Anzeigezeile,firstline=Dateizeile,lastline=Dateizeile]{Datei}`**: Einbinden eines Quelltexts aus einer Datei (alle Zeilenangaben sind optional)

Es folgt nun etwas Quellcode\ldots

```
\begin{lstlisting}
class Test
{
\end{lstlisting}
```

Es folgt nun etwas Quellcode...

```
1 class Test
2 {
```

Mathematikmodus

- TeX kennt Textmodus und Mathematikmodus
- **\$... \$** wechselt innerhalb einer Zeile in den Mathematikmodus, wobei Formeln eher kompakt gesetzt werden
- **\begin{equation} ... \end{equation}** erzeugt einen Absatz, in dem eine Formel steht (größzügiger gesetzt und nummeriert)
- Viele Befehle funktionieren nur im Mathematikmodus
- **^** stellt hoch, **_** stellt tief, mit besonderen Ergebnissen nach bestimmten Zeichen
- TeX kennt sehr viele Zeichen: <http://detexify.kirelabs.org/classify.html>

Die gaußsche Summenformel $\sum_{k=1}^n k = \frac{n^2 + n}{2}$
hier gleich nochmal:
$$\sum_{k=1}^n k = \frac{n^2 + n}{2}$$

Die gaußsche Summenformel
 $\sum_{k=1}^n k = \frac{n^2 + n}{2}$ hier gleich nochmal:
$$\sum_{k=1}^n k = \frac{n^2 + n}{2} \quad (1)$$

Querverweise

- **`\label{Name}`** definiert eine Marke, auf die aus dem Text verwiesen werden kann
 - Der Name muss eindeutig sein
 - **`\label{...}`** muss hinter dem Befehl stehen, auf den verwiesen werden soll, z.B.
`\section{...}\label{...}`
- **`\ref{Name}`** verweist auf eine Marke und setzt die Gliederungsnummer des Elements ein, das vor der Marke verwendet wurde
- **`\pageref{Name}`** setzt stattdessen die Seitenzahl ein
- Verweisen werden kann auf Überschriften, Abbildungen, Tabellen und Formeln

Literaturverzeichnis

- **`\bibliographystyle{ ... }`** legt in der Präambel den Stil der Literaturliste fest (**plain**, **abbrv**, **alpha**, **unsrt**, **apalike**)
 - Für deutsche Literaturlisten wird **`\usepackage{bibgerm}`** verwendet und üblicherweise **ger** vor den Stil geschrieben, z.B. **`\bibliographystyle{gerplain}`**
- Eine **.bib**-Datei enthält alle zitierbaren Literatureinträge, wobei jeder ein eindeutiges Kürzel hat, z.B. **`@inproceedings{mueller-11, ...`**
 - **Autor:innen** und **Editor:innen** werden durch **and** voneinander getrennt
- **`\cite{Kürzel}`** zitiert den Literatureintrag aus dem Text, z.B. **`\cite{mueller-11}`**
- **`\bibliography{Dateiname}`** erzeugt eine Literaturliste aus allen zitierten Einträgen, wobei der Dateiname die Endung **.bib** nicht enthält

Literaturverzeichnis

```
@inproceedings{mueller-11,  
  author = {Judith Müller and Tim Laue and Thomas Röfer},  
  title = {Kicking a Ball -- Modeling Complex Dynamic Motions for Humanoid Robots},  
  editor = {Javier Ruiz-del-Solar and Eric Chown and Paul G. Ploeger},  
  booktitle = {RoboCup 2010: Robot Soccer World Cup XIV},  
  publisher = {Springer},  
  volume = {6556},  
  pages = {109-120},  
  series = {Lecture Notes in Artificial Intelligence},  
  year = {2011}  
}
```

1. Müller, J., Laue, T., Röfer, T.: Kicking a ball – modeling complex dynamic motions for humanoid robots. In: del Solar, J.R., Chown, E., Ploeger, P.G. (eds.) RoboCup 2010: Robot Soccer World Cup XIV. Lecture Notes in Artificial Intelligence, vol. 6556, pp. 109–120. Springer (2011)

TeX-Dokumente bauen

- TeX besteht aus mehreren Compilern, die möglicherweise mehrfach in bestimmter Reihenfolge aufgerufen werden müssen (z.B. **pdfLaTeX**, **BibTeX**, **pdfLaTeX**, **pdfLaTeX**)
- Besser: Build-System nutzen
 - In einigen LaTeX-Editoren bereits integriert (z.B. in TeXstudio)
 - Ansonsten gibt es z.B. **latexmk** oder **texify**, die sich in Editoren einbinden lassen
- In TeX-Dateien ist die Groß-/Kleinschreibung von Dateinamen wichtig (**Windows** und **macOS** ignorieren sie, **Linux** aber nicht). Außerdem müssen **relative Pfade** verwendet werden
- Für PI-1 muss der LaTeX-Editor **UTF-8** als Zeichenkodierung verwenden (bei TeXstudio Standard)

Übungsblatt 1

- **zip**-Datei herunterladen und auspacken
- **loesung01.tex** so ergänzen, dass die erzeugte pdf-Datei so aussieht wie Vorgabe
- Keinen Java-Quellcode tippen, sondern aus **Primes.java** einbinden
- Verweise nutzen
- **Referenzen.bib** für Literaturliste nutzen

Praktische Informatik I WiSe 2022/23

Übungsblatt 1

Abgabe: nein

Aufgabe 1 Einmal mit \LaTeX , bitte

Erweitert die Datei *loesung01.tex* so, dass \LaTeX eine PDF-Ausgabedatei erzeugt, die der zweiten Seite dieses Übungsblatts gleicht. Einzig die Felder `<Tutor:in>` und `<Bearbeiter:in>` sollen durch die für euch passenden Daten ersetzt werden.

Folgende Konstrukte sollen eingesetzt werden:

- Hauptüberschriften
- Normaler Text
- Einbinden von externen Quelltextabschnitten. Verweist hierzu auf die beigelegte Datei *Primes.java*¹.
- Fetter Text
- Hervorgehobener Text. Dieser erscheint mit der verwendeten Vorlage kursiv.
- Querverweise
- Formelsatz
- Literaturverweis und Literaturliste mit BibTeX. Bindet hierzu die beigelegte Literaturliste *Referenzen.bib* ein und verwendet den Literaturstil *gerplain*.
- Fließende Umgebung mit Beschriftung
- Tabellensatz

Tipp. Beachtet, dass der Quelltext für Querverweise mehrfach übersetzt werden muss. Für das Literaturverzeichnis muss zusätzlich auch noch BibTeX ausgeführt werden (*pdflatex*, *bibtex*, *pdflatex*, *pdflatex*). Verwendet ihr *TeXstudio*, werden diese Schritte automatisch ausgeführt.

¹Wer möchte, kann das enthaltene Programm tatsächlich in BlueJ ausprobieren. Dazu die Datei *package.bluej* mit eben diesem öffnen. Über das Kontextmenü von *Primes* könnt ihr *void primes()* ausführen. Der gebogene Pfeil rechts unten im BlueJ-Fenster beendet das Programm wieder. Nichts davon ist aber für das Bearbeiten dieses Übungsblatts notwendig.