

myAssignment-Manual

Jasmin Sebastian

30. Oktober 2019

Die Klasse `myAssignment` stellt ein Layout zum Setzen von Übungszetteln zur Verfügung. Insbesondere wird ein vorgefertigter Header erstellt und es werden einige Abkürzungen und zusätzlichen Befehle zur Verfügung gestellt. Um die Präambel im Dokument möglichst kurz zu gestalten werden einige wichtige Pakete bereits geladen und Einstellungen z.B. für Codeumgebungen vorgenommen.

1 Optionenauswahl

Die `myAssignment`-Klasse bietet einige Möglichkeiten zur Optionenauswahl direkt beim Einbinden der Klasse, nach dem Schema `\documentclass[option]{myAssignment}`. Standardmäßig wird Schriftgröße *12* genutzt, dies kann hier geändert werden. Außerdem ist Deutsch als Sprache eingestellt, mit der Option *english* werden Überschriften etc. im Dokument auf Englisch umgestellt. Die Koma-Optionen *twocolumn* und *twoside* sind deaktiviert, ansonsten können alle Koma-Optionen wie gewohnt eingebunden und genutzt werden.

2 Layout

Um ein angepasstes Layout für Übungszettel zu bieten wird ein Header mit wichtigen Informationen erstellt. Dazu können der Name der Vorlesung, die Zettpartner, der Tutor, die Zettelnummer sowie das Semester über die Variablen `\title{Vorlesung}`, `\author{Zettpartner1, Zettpartner2}`, `\tutor{Tutor}`, `\nr{Zettelnummer}`, `\term{Semester}` eingestellt werden. Außerdem gibt es die Möglichkeit eine Punktetabelle für ein bis zehn Aufgaben auszugeben. Standardmäßig sind hier vier Aufgaben eingestellt, dies kann jedoch über den Befehl `\tasks{#Aufgaben}` geändert werden. Falls keine Nummer für die Anzahl der Aufgaben übergeben wird verschwindet die Tabelle, ein größerer Abstand bleibt allerdings vorhanden, dies kann verhindert werden, indem der Befehl mit dem

Keyword `none`, anstelle einer Zahl aufgerufen wird. Die Benennung der Überschriften wurde bereits auf *Aufgabe X* geändert, falls dies als Überschrift ausreichend ist, kann die Klammer bei `\section{}` einfach leer gelassen werden, ansonsten kann natürlich noch ein Überschrift hinzugefügt werden. Die `enumerate`-Umgebung wurde ebenfalls so angepasst, dass auf erster Ebene Kleinbuchstaben benutzt werden, sodass hierdurch Teilaufgaben gesetzt werden können.

3 Einstellungen und zusätzliche Befehle

Um Text zu Unterstreichen sollte anstelle von `\underline{text}`, entweder `\uline{text}` oder `\ul{text}` benutzt werden, da hier der Abstand zwischen Text und Unterstrich angepasst wurde. Um Code zu setzen kann die `lstlisting`-Umgebung benutzt werden, hierbei wurden schon einige Layouteinstellungen vorgenommen. So können nun auch \LaTeX -Befehle in der `lstlisting`-Umgebung ausgeführt werden, indem dieser Code escaped wird. Dies funktioniert über die escape-Sequenz `(/*text*/)`.

3.1 Mathesymbole und -umgebungen

Um das Arbeiten mit Mathesymbolen zu Vereinfachen wurden hier einige neue Befehle definiert um z.B. die Zahlenmengen $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$, sowie für Wahrscheinlichkeiten die Symbole $\mathcal{F}, \mathbb{P}, \mathbb{I}$ über die Befehle `\N, \Z, \Q, \R, \C, \F, \P, \one` darzustellen. Der Befehl `\equnder{equation}{subscript}` erzeugt ein vertikales Gleichzeichen unter einer Formel und das gegebene Ergebnis. Bei der `align*`-Umgebung haben die Formeln im Gegensatz zur `align`-Umgebung keine Nummerierung, durch den Befehl `\numberthis` kann hier für eine einzelne Zeile eine Nummer hinzugefügt werden. Außerdem werden Umgebungen für Definitionen, Sätze, Lemmata und Beispiele definiert, diese können über `\begin{mydef / satz / lemma / bsp} text \end{mydef / satz / lemma / bsp}` aufgerufen werden. Die zugehörigen Umgebung ohne Nummerierung können genauso mit einem `*` am Ende aufgerufen werden. Es können auch noch weiter Umgebungen über den Befehl `\newtheorem{name}[mydef]{Theoremname}` definiert werden. Der Teil `[mydef]` ist hier notwendig um die gleiche Nummerierung beizubehalten. Es wurde das Paket `amsopn` eingebunden um einige Matheoperatoren zur Verfügung zu stellen, so kann z.B. `\max()` im Mathemodus aufgerufen werden und wird dabei aufrecht geschrieben. Zusätzlich wurden `\argmax()` und `\argmin()` definiert. Eigene Matheoperatoren können außerdem über `\DeclareMathOperator{\op}{operator}` hinzugefügt werden.

3.2 Spaltentypen für Tabellen

Da Tabellenspalten die eine feste Breite haben sollen nur mit Blocksatz angeboten werden, wurden den bereits existierenden Spaltentypen noch drei weitere hinzugefügt. Diese können einfach über die Buchstaben *R* (rechtsbündige Spalte mit fester Breite), *L* (linksbündige Spalte mit fester Breite) und *C* (zentrierte Spalte mit fester Breite) eingebunden werden. Außerdem bietet der Spaltentyp *M* die Möglichkeit ganze Spalten im Mathemodus zu setzen.

3.3 Abkürzungen und weitere Befehle

Um Schreibarbeit zu sparen wurden für den Gebrauch im Textmodus die beiden Pfeile \rightarrow und \Rightarrow durch die Befehle `\ra` und `\Ra` abgekürzt. Der Befehl `\nix` fügt ein unsichtbares Zeichen an, was sinnvoll sein kann wenn man einen Abstand mit `\vspace{length}` erzeugen will, aber in der Zeile noch nichts steht.