

# lehramt-informatik

Hermine Bschlangaul \*

August 30, 2021

## Contents

<b>1</b>	<b>Klassen</b>	<b>3</b>
1.1	Vorlage Theorie-Teil	4
1.2	Vorlage Aufgabensammlung	5
1.3	Vorlage Aufgabe	6
<b>2</b>	<b>Pakete</b>	<b>7</b>
2.1	aufgaben-einbinden.sty	8
2.2	aufgaben-metadaten.sty	9
2.3	automaten.sty	10
2.3.1	Endlicher Automat	10
2.3.2	Kellerautomat	12
2.3.3	Turingmaschine	13
2.4	basis.sty	16
2.5	baum.sty	20
2.5.1	Binärbaum	21
2.5.2	AVL-Baum	22
2.5.3	B-Baum	23
2.6	checkbox.sty	24
2.7	chomsky-normalform.sty	25
2.8	cpm.sty	27
2.8.1	Hilfsmakros für Zeitpunkt-Tabelle	28
2.9	cyk-algorithmus.sty	30
2.10	entwurfsmuster.sty	31
2.10.1	Namensschema der Entwurfsmuster-Makros:	31
2.10.2	Reihenfolge	31
2.10.3	Abstrakte Fabrik (Abstract Factory)	31
2.10.4	Adapter	32
2.10.5	Beobachter (Observer)	34
2.10.6	Dekorierer (Decorator)	36
2.10.7	Einfache Fabrik (Simple Factory)	37
2.10.8	Einzelstück (Singleton)	38
2.10.9	Erbauer (Builder)	38
2.10.10	Fabrikmethode (Factory Method)	40
2.10.11	Kompositum (Composite)	41
2.10.12	Modell-Präsentation-Steuerung (Model-View-Controller)	42
2.10.13	Zustand (State)	43
2.11	er.sty	45
2.12	formale-sprachen.sty	47
2.13	formatierung.sty	50
2.13.1	Schriftarten / Typographie	50
2.13.2	Farben	50
2.13.3	Überschriften	50

---

\*E-mail: hermine.bschlangaul@gmx.net

2.13.4	Listen	50
2.13.5	Kasten	50
2.13.6	Header	50
2.14	gantt.sty	51
2.15	grafik.sty	52
2.16	graph.sty	53
2.17	hanoi.sty	55
2.18	komplexitaetstheorie.sty	56
2.19	kontrollflussgraph.sty	58
2.20	kopf-fusszeilen.sty	60
2.21	literatur-dummy.sty	61
2.22	literatur.sty	62
2.23	makros.sty	63
2.24	master-theorem.sty	67
2.25	mathe.sty	71
2.26	minimierung.sty	72
2.27	normalformen.sty	75
2.28	petri.sty	78
2.29	potenzmengen-konstruktion.sty	80
2.30	pseudo.sty	82
2.31	pumping-lemma.sty	83
2.32	quicksort.sty	84
2.33	relationale-algebra.sty	87
2.34	rmodell.sty	88
2.35	sortieren.sty	89
2.36	spalten.sty	91
2.37	struktogramm.sty	92
2.38	syntax.sty	93
2.39	syntaxbaum.sty	95
2.40	synthese-algorithmus.sty	96
2.41	tabelle.sty	99
2.42	typographie.sty	100
2.43	uml.sty	101
2.44	vollstaendige-induktion.sty	103
2.45	wasserfall.sty	105
2.46	wpkalkuel.sty	106

### 3 Index

107

# 1 Klassen

## 1.1 Vorlage Theorie-Teil

```
\documentclass{lehramt-informatik-haupt}

\begin{document}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Theorie-Teil
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

## 1.2 Vorlage Aufgabensammlung

```
\documentclass{lehramt-informatik-haupt}
\liLadeAllePakete

\begin{document}
\liAufgabe{30_AUD/06_Graphen/20_Dijkstra/Aufgabe_Graph-A-I}
\liAufgabe{30_AUD/06_Graphen/20_Dijkstra/Aufgabe_Graph-M-A-P-R-N}
\liAufgabe{30_AUD/06_Graphen/20_Dijkstra/Aufgabe_Staedte-A-F}
\liExamensAufgabe{46114/2008/09/Thema-1/Aufgabe-2}
\liExamensAufgabe{46115/2013/03/Thema-2/Aufgabe-5}
\liExamensAufgabe{66112/2004/03/Thema-1/Aufgabe-5}
\liExamensAufgabe{66115/2013/09/Thema-2/Aufgabe-9}
\liExamensAufgabe{66115/2015/03/Thema-2/Aufgabe-7}
\liExamensAufgabe{66115/2016/03/Thema-2/Aufgabe-6}
\liExamensAufgabe{66115/2017/03/Thema-1/Aufgabe-1}
\liExamensAufgabe{66115/2018/03/Thema-2/Aufgabe-9}
\liExamensAufgabe{66115/2020/09/Thema-1/Teilaufgabe-2/Aufgabe-3}
\end{document}
```

### 1.3 Vorlage Aufgabe

```
\documentclass{lehramt-informatik-aufgabe}
\liLadePakete{}
\begin{document}
\liAufgabenTitel{}
\section{
\index{DB}
\footcite{examen:}
}

\end{document}
```

## 2 Pakete

## 2.1 aufgaben-einbinden.sty

```
1 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2 \ProvidesPackage{lehramt-informatik-aufgaben-einbinden}[2020/06/13]

\liAufgabe Eine Aufgaben mit den Pfad relativ zu Wurzelverzeichnis dieses Repository einbinden.
3 \def\liAufgabe#1{
4   \input{\LehramtInformatikRepository/#1.tex}
5 }

\liExamensAufgabe Eine Examensaufgaben mit den Pfad relativ zu Staatsexamen einbinden, z. B.
\liExamensAufgabe{46116/2015/09/Thema-2/Teilaufgabe-1/Aufgabe-3}
6 \def\liExamensAufgabe#1{
7   \input{\LehramtInformatikRepository/Staatsexamen/#1.tex}
8 }

\liExamensAufgabeTTA
9 \def\liExamensAufgabeTTA #1 / #2 / #3 : Thema #4 Teilaufgabe #5 Aufgabe #6 {
10  \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Teilaufgabe-
    #5/Aufgabe-#6.tex}
11 }

\liExamensAufgabeTA
12 \def\liExamensAufgabeTA #1 / #2 / #3 : Thema #4 Aufgabe #5 {
13   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Aufgabe-#5.tex}
14 }

\liExamensAufgabeA
15 \def\liExamensAufgabeA #1 / #2 / #3 : Aufgabe #4 {
16   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Aufgabe-#4.tex}
17 }

18
```



## 2.2 aufgaben-metadaten.sty

```
19 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
20 \ProvidesPackage{lehramt-informatik-aufgaben-metadaten}[2020/07/07 Makros zum
21 Setzen der Aufgaben-Metadaten.]

22 \ExplSyntaxOn

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.
23 \def\liAufgabenTitel#1{
24 }

\liAufgabenMetadaten Setzen der Aufgaben-Metadaten über eine plist bzw. über key-values.
Die Schlüssel-Werte-Paare sind in der Datei basis.sty definiert. In der Typescript-
Datei .scripts/nodejs/src/aufgaben.ts gibt es ein entsprechendes Interface AufgabenMetadaten.

\liAufgabenMetadaten{
Titel = {Aufgabe 2},
Thematik = {Petri-Netz},
Stichwoerter = {Feld (Array), Implementierung in Java}
Fussnote = sosy:pu:4,
FussnoteSeite = {Seite 11},
BearbeitungsStand = OCR,
Korrektheit = absolut korrekt,
RelativerPfad = Staatsexamen/46116/2016/03/Thema-2/Teilaufgabe-1/Aufgabe-2.tex,
ExamenNummer = 46116,
ExamenJahr = 2016,
ExamenMonat = 03,
ExamenThemaNr = 2,
ExamenTeilaufgabeNr = 1,
ExamenAufgabeNr = 2,
}

25 \def\liAufgabenMetadaten#1{
26
27 \_setze_variablen_zurueck:
28
29 \tl_clear:N \g_auf_examen_telaufgabe_nr_tl
30
31 \keys_set:nn { aufgabenmetadaten } {
32 #1
33 }
34
35 \_setze_relativen_pfad:
36
37 \tl_if_empty:NTF \g_auf_titel_tl
38 {
39 \msg_fatal:nn { aufgabenmetadaten } { kein-titel }
40 }
41 {
42 }
43
44 \_gib_examen_titel: {}
45
46 \section{\_gib_aufgaben_titel:}
47
48 \bigskip
49 }

50 \ExplSyntaxOff

51
```

## 2.3 automaten.sty

```
52 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
53 \ProvidesPackage{lehramt-informatik-automaten}[2021/02/14 Zum Setzen von Automaten]
```

### 2.3.1 Endlicher Automat

```
54 \liLadePakete{formale-sprachen}

\begin{tikzpicture}[li automat]
\node[state,initial,accepting] (0) {$z_0$};
\node[state,right of=0] (1) {$z_1$};

\path (0) edge[above] node{1} (1);
\path (0) edge[loop,above] node{0} (0);
\path (1) edge[loop,above] node{0} (1);
\end{tikzpicture}
```



```
55 \RequirePackage{tikz}
56 \usetikzlibrary{arrows,automata,positioning}
57 \liLadePakete{mathe}
58 \directlua{
59   automaten = require('lehramt-informatik-automaten')
60 }
```

`\liAutomat` `\liAutomat[automaten-name]{zustaende=Z,alphabet=Σ,delta=δ,ende=E,start=z0}`

- `\liAutomat{}`:  $A = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat[A_1]{}`:  $A_1 = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat{zustaende={z_0, z_1, z_2}}`:  $A = (\{z_0, z_1, z_2\}, \Sigma, \delta, E, z_0)$
- `\liAutomat{alphabet={a,b}}`:  $A = (Z, \{a, b\}, \delta, E, z_0)$
- `\liAutomat{delta=d}`:  $A = (Z, \Sigma, d, E, z_0)$
- `\liAutomat{ende={z_0, z_1, z_2}}`:  $A = (Z, \Sigma, \delta, \{z_0, z_1, z_2\}, z_0)$
- `\liAutomat{start=z_1}`:  $A = (Z, \Sigma, \delta, E, z_1)$
- `\liAutomat{dea}`:  $A_{DEA} = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat{nea}`:  $A_{NEA} = (Z, \Sigma, \delta, E, z_0)$

```
61 \ExplSyntaxOn
62 \NewDocumentCommand {\liAutomat} { 0{A} m } {
63   \tl_set:Nn \l_zustaende_tl {Z}
64   \tl_set:Nn \l_alphabet_tl {\Sigma}
65   \tl_set:Nn \l_delta_tl {\delta}
66   \tl_set:Nn \l_ende_tl {E}
67   \tl_set:Nn \l_start_tl {z\sb{0}}
68   \tl_set:Nn \l_typ_tl {}
69
70   \keys_define:nn { automat } {
71     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
72     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
73     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
74     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
```

```

75     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
76     dea .value_forbidden:n = true,
77     dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
78     nea .value_forbidden:n = true,
79     nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
80   }
81
82   \keys_set:nn { automat } { #2 }
83
84   $#1 \l_typ_tl = (
85     \l_zustaende_tl,
86     \l_alphabet_tl,
87     \l_delta_tl,
88     \l_ende_tl,
89     \l_start_tl
90   )$
91 }
92 \ExplSyntaxOff

```

`\liAutomatenKante` **Let-Abkürzung:** `\let\k=\liAutomatenKante`

```

93 \def\liAutomatenKante#1#2#3#4{
94   \path (#1) edge[#4] node{#3} (#2);
95 }

96 \tikzset{
97   li automat/.style={
98     ->,
99     node distance=2cm
100   },
101 }

```

### 2.3.2 Kellerautomat

```
\begin{tikzpicture}[li kellerautomat,node distance=5cm]
\node[state,initial] (0) {\$z_0\$};
\node[state,right of=0,accepting] (1) {\$z_1\$};

\liKellerKante[above,loop]{0}{0}{
  1, KELLERBODEN, E KELLERBODEN;
  0, KELLERBODEN, KELLERBODEN;
}

\liKellerKante[above]{0}{1}{
  EPSILON, E, EPSILON;
  1, E, EPSILON;
  0, E, EPSILON;
}

\liKellerKante[above,loop]{1}{2}{
  EPSILON, KELLERBODEN, EPSILON;
  1, KELLERBODEN, KELLERBODEN;
  0, KELLERBODEN, KELLERBODEN;
}
\end{tikzpicture}
```



```
\liKellerAutomat \liKellerAutomat[\langle automaten-name \rangle]
{\zustaende=Z,alphabet=\Sigma,kelleralphabet=\Gamma,delta=\delta,start=z_0,kellerboden=\#,ende=E}

\liKellerAutomat{
  zustaende={z_0, z_1, z_2},
  alphabet={a, b, c},
  kelleralphabet={\#, A},
  ende={z_2},
}
```

$$K = (\{z_0, z_1, z_2\}, \{a, b, c\}, \{\#, A\}, \delta, z_0, \#, \{z_2\})$$

```
102 \ExplSyntaxOn
103 \NewDocumentCommand {\liKellerAutomat} { O{K} m } {
104   \tl_set:Nn \l_zustaende_tl {Z}
105   \tl_set:Nn \l_alphabet_tl {\Sigma}
106   \tl_set:Nn \l_kelleralphabet_tl {\Gamma}
107   \tl_set:Nn \l_delta_tl {\delta}
108   \tl_set:Nn \l_start_tl {z\sb{0}}
109   \tl_set:Nn \l_kellerboden_tl {\#}
110   \tl_set:Nn \l_ende_tl {E}
111
112   \keys_define:nn { kellerautomat } {
113     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
114     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
115     kelleralphabet .code:n = {\tl_set:Nn \l_kelleralphabet_tl {\liMenge{##1}}},
116     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
117     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
```

```

118     kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
119     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
120   }
121
122   \keys_set:nn { kellerautomat } { #2 }
123
124   $#1 = (
125     \l_zustaende_tl,
126     \l_alphabet_tl,
127     \l_kelleralphabet_tl,
128     \l_delta_tl,
129     \l_start_tl,
130     \l_kellerboden_tl,
131     \l_ende_tl
132   )$
133 }
134 \ExplSyntaxOff

```

**\liKellerUebergang** **Let-Abkürzung:** `\let\u=\liKellerUebergang`  
`\liKellerUebergang{a, KELLERBODEN, A; b KELLERBODEN, EPSILON}`

(a, #: A)  
(b, #:  $\varepsilon$ )

Regulärer Ausdruck zum Konvertieren: `\((.*)`, `(.*)`, `(.*)\)` `\u{$1 $2 $3}`

```

135 \ExplSyntaxOn
136 \def\liKellerUebergang#1{
137   \directlua{automaten.drucke_keller_uebergaenge('#1')}
138 }
139 \ExplSyntaxOff

```

**\liKellerKante** `\liKellerKante[\tikz-optionen]{\von}{\zu}{\übergange}`  
**Let-Abkürzung:** `\let\k=\liKellerKante`

```

140 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
141   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
142 }

143 \tikzset{
144   li keller knoten/.style={
145     text width=2cm,
146     align=center,
147     font=\footnotesize,
148   },
149   li kellerautomat/.style={
150     li automat,
151     every edge/.append style={
152       every node/.style={
153         li keller knoten
154       }
155     }
156   }
157 }

```

### 2.3.3 Turingmaschine

```
158 \RequirePackage{amssymb}
```

**\liTuringLeerzeichen**

□

```
159 \def\liTuringLeerzeichen{\Box}
```

```

\liTuringMaschine \liTuringMaschine[\langle automaten-name \rangle]
{\zustaende=Z,alphabet= $\Sigma$ ,bandalphabet= $\Gamma$ ,delta= $\delta$ ,start= $z_0$ ,leerzeichen= $\square$ ,ende=E)}

\liTuringMaschine{
  zustaende={z_0, z_1, z_2},
  alphabet={a, b, c},
  bandalphabet={\liTuringLeerzeichen, A},
  ende={z_2},
}

```

$TM = (\{z_0, z_1, z_2\}, \{a, b, c\}, \{\square, A\}, \delta, z_0, \square, \{z_2\})$
--

```

160 \ExplSyntaxOn
161 \NewDocumentCommand {\liTuringMaschine} { O{TM} m } {
162   \tl_set:Nn \l_zustaende_tl {Z}
163   \tl_set:Nn \l_alphabet_tl {\Sigma}
164   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
165   \tl_set:Nn \l_delta_tl {\delta}
166   \tl_set:Nn \l_start_tl {z\sb{0}}
167   \tl_set:Nn \l_leerzeichen_tl {\liTuringLeerzeichen}
168   \tl_set:Nn \l_ende_tl {E}
169
170   \keys_define:nn { kellerautomat } {
171     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
172     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
173     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\liMenge{##1}}},
174     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
175     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
176     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
177     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
178   }
179
180   \keys_set:nn { kellerautomat } { #2 }
181
182   $\text{#1} = (
183     \l_zustaende_tl,
184     \l_alphabet_tl,
185     \l_bandalphabet_tl,
186     \l_delta_tl,
187     \l_start_tl,
188     \l_leerzeichen_tl,
189     \l_ende_tl
190   )$
191 }
192 \ExplSyntaxOff

```

**\liTuringUebergangZelle** Formatiert einen Zustandsübergang für eine Übergangstabelle.

**Let-Abkürzung:** `\let\t=\liTuringUebergangZelle`

`\liTuringUebergangZelle{z_1, LEER, R}:(z_1:  $\square$ , R) \liTuringUebergangZelle{z1, leer, l}:`  
`(z1:  $\square$ , L)`

```

193 \ExplSyntaxOn
194 \def\liTuringUebergangZelle#1{
195   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
196 }
197 \ExplSyntaxOff

```

**\liTuringUebergaenge** Nur in den TikZ-Grafiken zu verwenden. Setzt Zeilenumbrüche ans Ende. Nicht für die Tabelle geeignet.

**Let-Abkürzung:** `\let\t=\liTuringUebergaenge`

`(z1:  $\square$ , L)`

`( $\square$ :  $\square$ , R)`

```

198 \ExplSyntaxOn
199 \def\liTuringUebergaenge#1{
200   \directlua{automaten.drucke_turing_uebergaenge('#1')}
201 }
202 \ExplSyntaxOff

\liTuringKante \liTuringKante[\tikz-optionen]{\langle zustand-oder-lese \rangle}{\langle schreibe \rangle}{\langle richtung \rangle}
Let-Abkürzung: \let\t=\liTuringKante

203 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
204   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
205 }

\liTuringUeberfuehrung

206 \def\liTuringUeberfuehrung{
207    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
208 }

209 \tikzset{
210   li turingmaschine/.style={
211     li automat,
212     every edge/.append style={
213       every node/.style={
214         li keller knoten
215       }
216     }
217   }
218 }
219

```

## 2.4 basis.sty

```
220 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
221 \ProvidesPackage{lehramt-informatik-basis}[2020/11/27]

222 \RequirePackage{xparse}

223 \ExplSyntaxOn

\liLadePakete

224 \NewDocumentCommand{\liLadePakete}{ m }
225 {
226   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
227 }

\liLadeAllePakete

228 \def\liLadeAllePakete{
229   \liLadePakete{
230     aufgaben-einbinden,
231     automaten,
232     baum,
233     checkbox,
234     chomsky-normalform,
235     cpm,
236     cyk-algorithmus,
237     entwurfsmuster,
238     er,
239     formale-sprachen,
240     gantt,
241     grafik,
242     graph,
243     hanoi,
244     kontrollflussgraph,
245     makros,
246     master-theorem,
247     mathe,
248     minimierung,
249     normalformen,
250     petri,
251     potenzmengen-konstruktion,
252     pseudo,
253     quicksort,
254     relationale-algebra,
255     rmodell,
256     sortieren,
257     spalten,
258     struktogramm,
259     syntax,
260     syntaxbaum,
261     synthese-algorithmus,
262     tabelle,
263     typographie,
264     uml,
265     vollstaendige-induktion,
266     wasserfall,
267     wpkalkuel,
268   }
269 }
```

Definition einer Komma-getrennten-Liste mit deren Hilfe die vielen globalen Token-List-Variablen definiert werden können. Die einzelnen Schlüssel sind im Interface AufgabenMetadaten in der Typescript-Datei `.scripts/nodejs/src/aufgaben.ts` erklärt.

```
270 \clist_new:N \g_auf_schluesssel_clist
271 \clist_set:Nn \g_auf_schluesssel_clist {
```



```

272  titel,
273  thematik,
274  stichwoerter,
275  fussnote,
276  fussnote_seite,
277  %
278  bearbeitungs_stand,
279  korrektheit,
280  %
281  relativer_pfad,
282  %
283  examen_nummer,
284  examen_jahr,
285  examen_monat,
286  examen_thema_nr,
287  examen_telaufgabe_nr,
288  examen_aufgabe_nr,
289 }

```

Initialisierung der globalen Token-List-Variablen `\g_auf_***_tl`. `auf` steht für Aufgabe.

```

290 \clist_map_inline:Nn \g_auf_schluesssel_clist {
291   \tl_new:c {g_auf_#1_tl}
292 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

293 \cs_new:Npn \setze_variablen_zurueck: {
294   \clist_map_inline:Nn \g_auf_schluesssel_clist {
295     \tl_clear:c {g_auf_##1_tl}
296   }
297 }

```

Die einzelnen Schlüssel sind im Interface `AufgabenMetadaten` in der Typescript-Datei `.scripts/nodejs/src/aufgaben.ts` erklärt.

```

298 \keys_define:nn { aufgabenmetadaten }
299 {
300   Titel .tl_gset:N = \g_auf_titel_tl,
301   Thematik .tl_gset:N = \g_auf_thematik_tl,
302   Stichwoerter .tl_gset:N = \g_auf_stichwoerter_tl,
303   Fussnote .tl_gset:N = \g_auf_fussnote_tl,
304   FussnoteSeite .tl_gset:N = \g_auf_fussnote_seite_tl,
305   %
306   BearbeitungsStand .tl_gset:N = \g_auf_bearbeitungs_stand_tl,
307   Korrektheit .tl_gset:N = \g_auf_korrekttheit_tl,
308   %
309   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
310   %
311   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
312   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
313   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
314   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
315   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_telaufgabe_nr_tl,
316   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
317 }

318 \cs_gset:Npn \setze_relativen_pfad: {
319   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
320   {
321     \bool_if:nTF
322     {
323       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
324       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
325       ! \tl_if_empty_p:N \g_auf_examen_monat_tl
326     }
327     {

```

```

328 \tl_gset:Nn \g_auf_relativer_pfad_tl {
329   Staatsexamen /
330   \g_auf_examen_nummer_tl /
331   \g_auf_examen_jahr_tl /
332   \g_auf_examen_monat_tl /
333   \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
334   \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_te
335   \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr
336 }
337 }
338 {}
339 }
340 {}
341 }

342 \cs_gset:Npn \_gib_examen_titel: {
343   \cs_set:Nn \_trenner: { \, / \, }
344   \bool_if:nTF
345   {
346     ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
347     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
348     ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
349     ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
350   }
351   {
352     {
353       \footnotesize
354       \par
355       \noindent
356       Staatsexamen ~
357       \g_auf_examen_nummer_tl \_trenner:
358       \g_auf_examen_jahr_tl \_trenner:
359
360       \tl_case:Nn \g_auf_examen_monat_tl
361       {
362         { 03 } { Frühjahr }
363         { 09 } { Herbst }
364       } \_trenner:
365
366       \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
367         Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
368       }
369       \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
370         Teilaufgabe ~ Nr. ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
371       }
372       \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
373         Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
374       }
375       \par
376       \bigskip
377     }
378   }
379 }

380 \cs_new:Npn \_gib_github_url: {
381   \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
382     Github :~\href{
383       \LehramtInformatikGithubRawDomain /
384       \LehramtInformatikGithubTexRepo /
385       \LehramtInformatikGitBranch /
386       \g_auf_relativer_pfad_tl
387     }{
388       \g_auf_relativer_pfad_tl
389     }
390   }

```

```

391 }
392 \cs_new:Npn \_gib_aufgaben_titel: {
393   \g_auf_titel_tl
394
395   \tl_if_empty:NTF \g_auf_thematik_tl
396   {}
397   {
398     \, ~ [
399       \g_auf_thematik_tl
400     ]
401   }
402 }
403 \msg_new:nnn { aufgabenmetadaten } { kein-titel }
404 { Der~Schlüssel~ist~zwingend~notwendig }
405 \ExplSyntaxOff
406 \def\LehramtInformatikRepository{/pfad/zum/repository}
407 \input /etc/lehramt-informatik.config.tex
408 biblatex not working with lualatex and babel
409 % \RequirePackage{polyglossia}
410 % \setmainlanguage{german}

```

## 2.5 baum.sty

```
410 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
411 \ProvidesPackage{lehramt-informatik-baum}[2020/06/13 Zum Setzen von
412 Binär- und AVL-Bäumen. Hüll-Paket um TikZ and tikz-qtree.]

413 \RequirePackage{tikz}

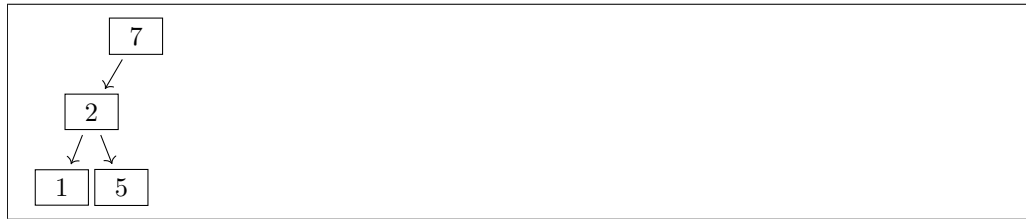
    für li binaer baum
414 \RequirePackage{tikz-qtree}

    Für b baum
415 \usetikzlibrary{shapes.multipart}
```

### 2.5.1 Binärbaum

TikZ-Stil: li binaer baum: Knoten als Kreise, Kanten als Pfeile.

```
\begin{tikzpicture}[li binaer baum]
\Tree
[.7
  [.2
    [.1 ]
    [.5 ]
  ]
\edge[blank]; \node[blank]{};
]
\end{tikzpicture}
```



```
416 \tikzset{
417   li binaer baum/.style={
418     shorten <=2pt,
419     shorten >=2pt,
420     ->,
421     every tree node/.style={
422       minimum width=2em,
423       draw,
424       rectangle
425     },
426     blank/.style={
427       draw=none
428     },
429     edge from parent/.style={
430       draw,
431       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
432     },
433     level distance=1cm,
434     every label/.style={
435       gray,
436       font=\footnotesize,
437       label position=0,
438       label distance=0cm,
439     }
440   },
441 }
```

### 2.5.2 AVL-Baum

```
\begin{tikzpicture}[li binaer baum]
\Tree
[.\node[label=-1]{5};
  [.\node[label=-1]{2};
    [.\node[label=0]{1}; ]
    \edge[blank]; \node[blank]{};
  ]
  [.\node[label=0]{7}; ]
]
\end{tikzpicture}
```



### 2.5.3 B-Baum

```

\begin{tikzpicture}[
  scale=0.8,
  transform shape,
  li bbaum,
  level 1/.style={level distance=10mm,sibling distance=32mm},
  level 2/.style={level distance=10mm,sibling distance=20mm},
]
\node {10 \nodepart{two} 33}
  child {node {8}
    child {node {5}}
    child {node {9}}
  }
  child {node {20}
    child {node {15}}
    child {node {31}}
  }
  child {node {50}
    child {node {45}}
    child {node {60 \nodepart{two} 80}}
  }
;
\end{tikzpicture}

```



```

442 \tikzset{
443   li bbaum knoten/.style={
444     rectangle split parts=10,
445     rectangle split,
446     rectangle split horizontal,
447     rectangle split ignore empty parts,
448     draw,
449     fill=white
450   },
451   li bbaum/.style={
452     every node/.style={
453       li bbaum knoten
454     },
455     level 1/.style={
456       level distance=12mm,
457       sibling distance=25mm,
458     },
459     every child/.style={
460       shorten <= 2pt,
461       shorten >= 6pt,
462       ->,
463     },
464     level 2/.style={
465       level distance=9mm,
466       sibling distance=15mm,
467     },
468   }
469 }
470

```

## 2.6 checkbox.sty

```
471 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
472 \ProvidesPackage{lehramt-informatik-checkbox}[2020/12/14 Zum Setzen von
473 Multiple-Choice-Fragen. Simulation von Kästchen zum ankreuzen]
474 \RequirePackage{amssymb}

\liRichtig Angekreuztes Kästchen (nur innerhalb der itemize-Umgebung zu verwenden.
475 \def\liRichtig{\item[$\boxtimes$]}

\liFalsch Nicht angekreuztes Kästchen (nur innerhalb der itemize-Umgebung zu verwenden.
476 \def\liFalsch{\item[$\square$]}

477
```



## 2.7 chomsky-normalform.sty

```

478 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
479 \ProvidesPackage{lehramt-informatik-chomsky-normalform}[2021/03/26
480 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]
481
482 \ExplSyntaxOn
483
484 \liLadePakete{typographie}

\liChomskyUeberschrift \Let-Abkürzung: \let\schrift=\liChomskyUeberschrift

485 \def\liChomskyUeberschrift#1{
486   {
487     \bfseries
488     \sffamily
489     \str_case:nn {#1} {
490       {1} {Elimination~der~ $\varepsilon$ -Regeln}
491       {2} {Elimination~von~Kettenregeln}
492       {3} {Separation~von~Terminalzeichen}
493       {4} {Elimination~von~mehrelementigen~Nonterminalketten}
494     }
495   }
496 }

\liChomskyErklaerung \Let-Abkürzung: \let\erklaerung=\liChomskyErklaerung
Hoffmann Seite 180

497 \def\li@chomsky@erklaerung@texte#1{
498   \str_case:nn {#1} {
499     %
500     {1} {
501       Alle~Regeln~der~Form~ $A \rightarrow \varepsilon$ ~werden~eliminiert.~
502       Die~Ersetzung~von~ $A$ ~wird~durch~ $\varepsilon$ ~in~allen~anderen~
503       Regeln~vorweggenommen.
504     }
505     {2} {
506       Jede~Produktion~der~Form~ $A \rightarrow B$ ~mit~ $A, B \in S$ ~wird~
507       als~Kettenregel~bezeichnet.~Diese~tragen~nicht~zur~Produktion~
508       von~Terminalzeichen~bei~und~lassen~sich~ebenfalls~eliminieren.
509     }
510     {3} {
511       Jedes~Terminalzeichen~ $\sigma$ ,~das~in~Kombination~mit~anderen~
512       Symbolen~auftaucht,~wird~durch~ein~neues~Nonterminal~
513        $S_{\sigma}$ ~ersetzt~und~die~Menge~der~Produktionen~durch~die~
514       Regel~ $S_{\sigma} \rightarrow \sigma$ ~ergänzt.
515     }
516     {4} {
517       Alle~Produktionen~der~Form~
518        $A \rightarrow B_1 B_2 \dots B_n$ ~
519       werden~in~die~Produktionen~
520        $A \rightarrow B_1 B_2 \dots B_{n-1} B_n$ ~
521        $A \rightarrow B_1 B_2 \dots B_{n-1} B_n$ ~
522        $A \rightarrow B_1 B_2 \dots B_{n-1} B_n$ ~
523        $A \rightarrow B_1 B_2 \dots B_{n-1} B_n$ ~zerteilt.~
524       Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
525       vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
526     }
527   }
528 }

529 \def\liChomskyErklaerung#1{
530   {
531     \itshape
532     \footnotesize
533     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
534   }

```

```
535 }
```

```
\liChomskyUeberErklaerung Let-Abkürzung: \let\schrittE=\liChomskyUeberErklaerung
```

```
536 \def\liChomskyUeberErklaerung#1{
```

```
537   \liChomskyUeberschrift{#1}\par
```

```
538   \liChomskyErklaerung{#1}
```

```
539 }
```

```
540 \ExplSyntaxOff
```

```
541
```

## 2.8 cpm.sty

```

542 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
543 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
544 \RequirePackage{tikz}
545 \liLadePakete{mathe}

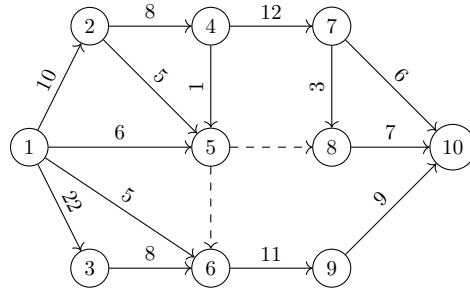
```

### Let-Abkürzungen

```

\let\f=\footnotesize
\let\FZ=\liCpmFruehesterI
\let\SZ=\liCpmSpaetesterI
\let\v=\liCpmVon
\let\vz=\liCpmVonZu
\let\z=\liCpmZu

```



```

\begin{tikzpicture}[scale=0.8,transform shape]
\liCpmEreignis{1}{0}{2}
\liCpmEreignis{2}{1}{4}
\liCpmEreignis{3}{1}{0}

\liCpmVorgang{1}{2}{10}
\liCpmVorgang{1}{3}{22}
\liCpmVorgang{1}{5}{6}

\liCpmVorgang[schein]{5}{6}{5}
\liCpmVorgang[schein]{5}{8}{7}
\end{tikzpicture}

```

```

\begin{tabular}{|l|l|l|l|l|l|l|l|}
\hline
$i$ & a & b & c & d & e & f & g \\
\hline
\FZ & 0 & 2 & 5 & 6 & 10 & 3 & 12 \\
\hline
\SZ & 0 & 2 & 5 & 10 & 10 & 6 & 12 \\
\hline
GP & 0 & 0 & 0 & 3 & 0 & 3 & 0 \\
\hline
\end{tabular}

```

```

\liCpmEreignis liCpmEreignis\{(.*)\}\{((.*),(.*))\} -> liCpmEreignis{$1}{$2}{$3}

```

```

546 \ExplSyntaxOn
547 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
548   \tl_set:Nn \l_name_tl {}
549
550   \keys_define:nn { cpmEreignis } {
551     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
552   }
553
554   \keys_set:nn { cpmEreignis } { #1 }
555
556   \tl_if_empty:NT \l_name_tl {
557     \tl_set:Nn \l_name_tl {#2}
558   }

```

```

559
560 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
561 }
562 \ExplSyntaxOff

\liCpmVorgang \liCpmVORGANG\((.*)>(.*))\{(.*)\}
563 \ExplSyntaxOn
564 \NewDocumentCommand { \liCpmVorgang } { 0{ } m m m } {
565   \tl_set:Nn \l_schein_tl {}
566   \tl_set:Nn \l_kritisch_tl {}
567
568   \keys_define:nn { cpmVorgang } {
569     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
570     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
571   }
572
573   \keys_set:nn { cpmVorgang } { #1 }
574
575   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
576 }
577 \ExplSyntaxOff

```

### 2.8.1 Hilfsmakros für Zeitpunkt-Tabelle

```

\begin{tabular}{|l||l|l|l|l|l|l|l|l|}
\hline
\hline
\hline
\hline
\hline
\hline
\hline
\hline
\hline
\end{tabular}

```

```

\liCpmVonZu Let-Abkürzung: \let\vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $l_{(2 \rightarrow 3)}$ 
578 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
579 \def\liCpmVonZu#1(#2-#3){%
580   \ifmmode%
581     \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
582   \else%
583     $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
584   \fi%
585 }

```

```

\liCpmVon Let-Abkürzung: \let\v=\liCpmVon
\liCpmVon{1}(2):  $l_{(\rightarrow 2)}$ 
586 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
587 \def\liCpmVon#1(#2){%
588   \ifmmode%
589     \liCpmVonOhneMathe{#1}(\#2)%
590   \else%
591     $\liCpmVonOhneMathe{#1}(\#2)$%
592   \fi%
593 }

```

```

\liCpmZu Let-Abkürzung: \let\z=\liCpmZu
\liCpmZu{1}(2):  $l_{(\leftarrow 2)}$ 
594 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
595 \def\liCpmZu#1(#2){%
596   \ifmmode%
597     \liCpmZuOhneMathe{#1}(\#2)%
598   \else%
599     $\liCpmZuOhneMathe{#1}(\#2)$%

```

```

600 \fi%
601 }

\liCpmSpaetesterI Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann
Let-Abkürzung: \let\SZ=\liCpmSpaetesterI
602 \def\liCpmSpaetesterI{$SZ_i$}

\liCpmFruehesterI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.
Let-Abkürzung: \let\FZ=\liCpmFruehesterI
603 \def\liCpmFruehesterI{$FZ_i$}

604

```

## 2.9 cyk-algorithmus.sty

```
605 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
606 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
607 zum Setzen des CYK-Algorithmus]
```

```
\begin{tabular}{|c|c|c|c|c|c|}
a      & c      & b      & c      & a      & b \\ \hline \hline

$R_a$ & $R_c$ & $R_b$ & $R_c$ & $R_a$ & $R_b$ \16
B      & A      & A      & B      & C \15
S      & -      & S      & S \14
-      & -      & - \13
-      & - \12
S \11
\end{tabular}
```

```
\liKurzeTabellenLinie Let-Abkürzung: \let\l=\liKurzeTabellenLinie
608 \def\liKurzeTabellenLinie#1{\ccline{1-#1}}
```

```
\liWortInSprache  $\Rightarrow abc \in L(Y)$ 
609 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
610   \bigskip
611   \noindent
612    $\Rightarrow$  #1 \in #2$
613 }
```

```
\liWortNichtInSprache  $\Rightarrow abc \notin L(G)$ 
614 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
615   \bigskip
616   \noindent
617    $\Rightarrow$  #1 \notin #2$
618 }

619
```

## 2.10 entwurfsmuster.sty

```
620 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
621 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06
622 Hilfsmakros zum Setzen von Entwurfsmustern / Design Patterns]
```

### 2.10.1 Namensschema der Entwurfsmuster-Makros:

Präfix: \liEntwurfs + Name des Entwurfsmuster DeutscherName + Suffix: (Uml, Akteure, Code, ohne)

### 2.10.2 Reihenfolge

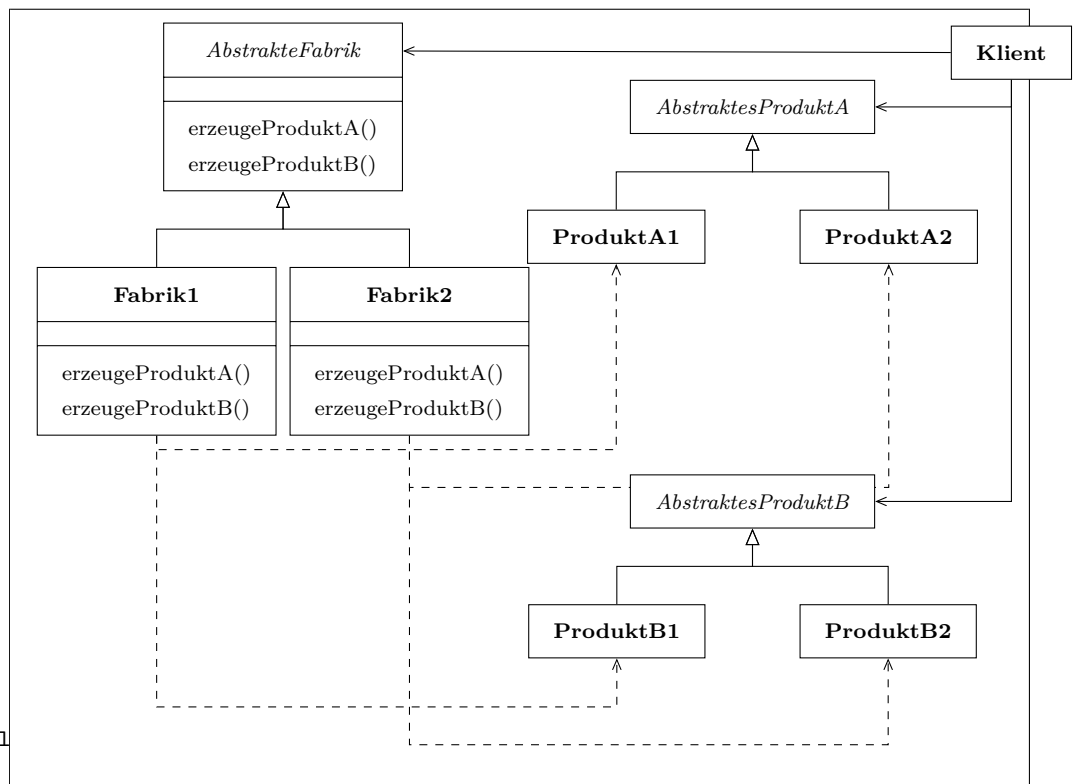
1. Uml: Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. Akteure: Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. Code: Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

```
623 \RequirePackage{lehramt-informatik-uml}
```

`\li@EntwurfsCodeAllgemein` Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
624 \RequirePackage{lehramt-informatik-syntax}
625 \def\li@EntwurfsCode#1#2{
626   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
627 }
```

### 2.10.3 Abstrakte Fabrik (Abstract Factory)



```

628 \def\liEntwurfsAbstrakteFabrikUml{
629   \begin{tikzpicture}
630     \umlclass[type=abstract]{AbstrakteFabrik}{-}{
631       erzeugeProduktA())\\
632       erzeugeProduktB())\\
633     }

```

```

634 \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
635     erzeugeProduktA()\
636     erzeugeProduktB()\
637 }
638 \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
639     erzeugeProduktA()\
640     erzeugeProduktB()\
641 }
642 \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
643 \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
644
645 \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
646 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
647 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
648 \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
649 \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
650
651 \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
652
653 \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
654 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
655 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
656 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
657 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
658
659 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
660 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
661
662 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
663 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
664
665 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
666 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
667 \umlVHuniassoc{Klient}{AbstraktesProduktB}
668 \end{tikzpicture}
669 }

```

iEntwurfsAbstrakteFabrikCode

```

670 \def\liEntwurfsAbstrakteFabrikCode{
671     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
672     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
673     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
674 }

```

\liEntwurfsAbstrakteFabrik

```

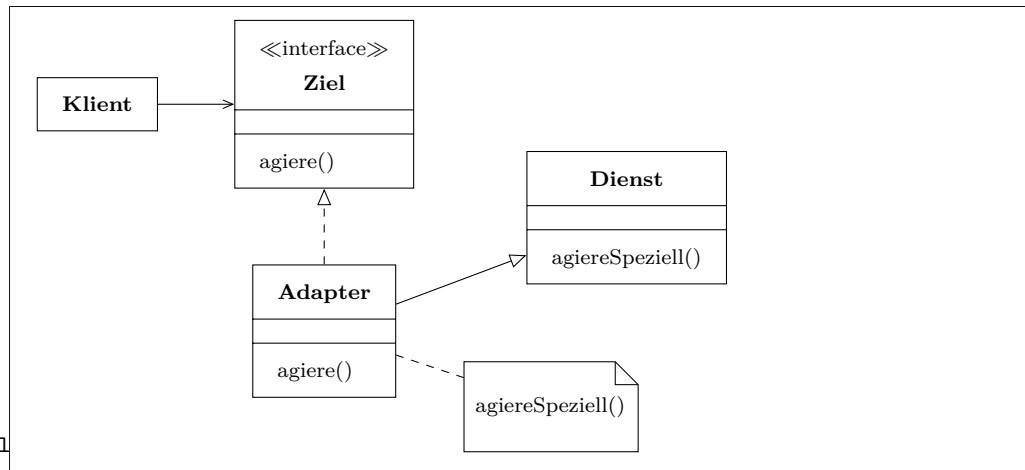
675 \def\liEntwurfsAbstrakteFabrik{
676     \liEntwurfsAbstrakteFabrikUml
677     \liEntwurfsAbstrakteFabrikCode
678 }

```

## 2.10.4 Adapter



\liEntwurfsAdapterUml



```

679 \def\liEntwurfsAdapterUml{
680   \begin{tikzpicture}
681     \umlsimpleclass[x=1,y=3]{Klient}{}{}
682     \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
683     \umlclass[x=4,y=0]{Adapter}{}{agiere()}
684     \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
685
686     \umlreal{Adapter}{Ziel}
687     \umluniassoc{Klient}{Ziel}
688     \umlinherit{Adapter}{Dienst}
689
690     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
691   \end{tikzpicture}
692   \footcite[so ähnlich wie GoF]{wiki:adapter}
693 }

```

\liEntwurfsAdapterAkteure

**Ziel (Target)** Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.

**Klient (Client)** Der Klient nutzt Dienste über inkompatible Schnittstellen und greift dabei auf adaptierte Schnittstellen zurück.

**Dienst (Adaptee)** Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest definierter Schnittstelle an.

**Adapter** Der Adapter adaptiert die Schnittstelle des Dienstes auf die Schnittstelle zum Klienten.

```

694 \def\liEntwurfsAdapterAkteure{
695   \begin{description}
696
697     \item[Ziel (Target)]
698
699     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
700
701     \item[Klient (Client)]
702
703     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
704     dabei auf adaptierte Schnittstellen zurück.
705
706     \item[Dienst (Adaptee)]
707
708     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
709     definierter Schnittstelle an.
710
711     \item[Adapter]
712
713     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
714     Schnittstelle zum Klienten.\footcite{wiki:adapter}

```

```

715
716 \end{description}
717 }

```

\liEntwurfsAdapterCode

```

718 \def\liEntwurfsAdapterCode{
719 \li@EntwurfsCode{adapter}{Dienst}
720 \li@EntwurfsCode{adapter}{Ziel}
721 \li@EntwurfsCode{adapter}{Adapter}
722 \li@EntwurfsCode{adapter}{Klient}
723 }

```

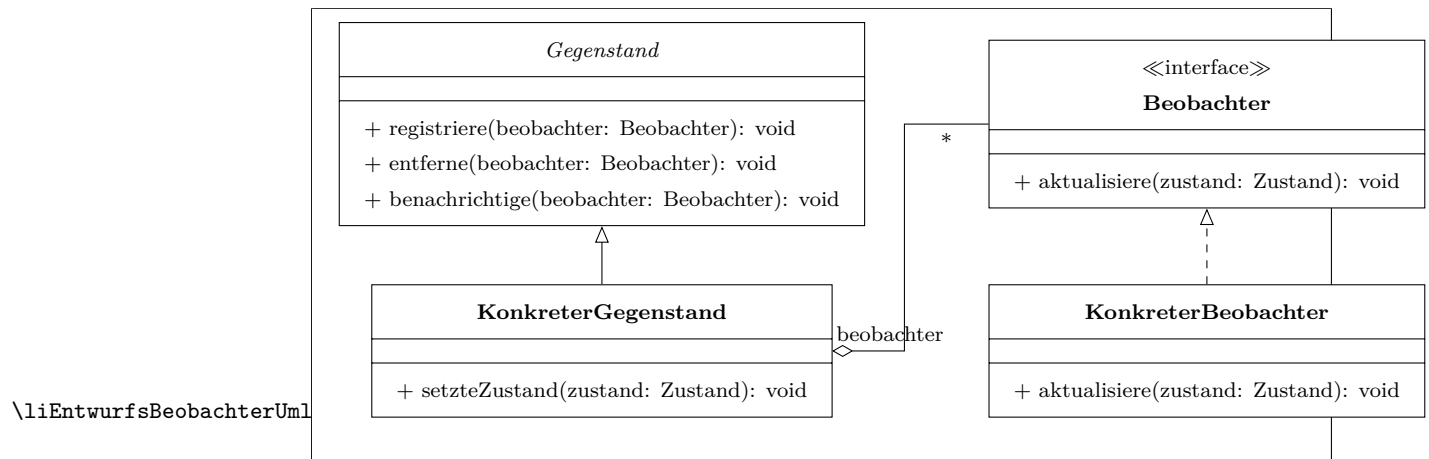
\liEntwurfsAdapter

```

724 \def\liEntwurfsAdapter{
725 \liEntwurfsAdapterUml
726 \liEntwurfsAdapterAkteure
727 \liEntwurfsAdapterCode
728 }

```

## 2.10.5 Beobachter (Observer)



```

729 \def\liEntwurfsBeobachterUml{
730 \begin{tikzpicture}
731 \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
732 + registriere(beobachter: Beobachter): void\\
733 + entferne(beobachter: Beobachter): void\\
734 + benachrichtige(beobachter: Beobachter): void\\
735 }
736 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
737 + setzteZustand(zustand: Zustand): void
738 }
739 \umlinherit{KonkreterGegenstand}{Gegenstand}
740
741 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
742 + aktualisiere(zustand: Zustand): void
743 }
744 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
745 + aktualisiere(zustand: Zustand): void
746 }
747 \umlreal{KonkreterBeobachter}{Beobachter}
748
749 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
750 {KonkreterGegenstand}{Beobachter}
751 \end{tikzpicture}
752 }

```

\liEntwurfsBeobachterAkteure

**Gegenstand / Subjekt (Subject / Observable)** Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An- und Abmeldung von Beobachtern und eine Schnittstelle zur Benachrichtigung von Beobachtern über Änderungen an.

**Beobachter (Observer)** Die Beobachter (auf Englisch auch subscriber, also „Abonnent“, genannt) definieren eine Aktualisierungsschnittstelle.

**konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)**

Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert den relevanten Zustand und benachrichtigt alle Beobachter bei Zustandsänderungen über deren Aktualisierungsschnittstelle. Es verfügt über eine Schnittstelle zur Erfragung des aktuellen Zustands.

**Konkrete Beobachter (ConcreteObserver)** Konkrete Beobachter verwalten die Referenz auf ein konkretes Subjekt, dessen Zustand sie beobachten und speichern und dessen Zustand konsistent ist. Sie implementieren eine Aktualisierungsschnittstelle unter Verwendung der Abfrageschnittstelle des konkreten Subjekts.

```

753 \def\liEntwurfsBeobachterAkteure{
754   \begin{description}
755     \item[Gegenstand / Subjekt (Subject / Observable)]
756
757     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
758     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
759     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
760     und Abmeldung von Beobachtern und eine Schnittstelle zur
761     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
762     251]{gof}
763
764     \item[Beobachter (Observer)]
765
766     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
767     genannt) definieren eine Aktualisierungsschnittstelle.
768
769     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
770
771     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
772     den relevanten Zustand und benachrichtigt alle Beobachter bei
773     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
774     verfügt über eine Schnittstelle zur Erfragung des aktuellen
775     Zustands.
776
777     \item[Konkrete Beobachter (ConcreteObserver)]
778
779     Konkrete Beobachter verwalten die Referenz auf ein konkretes
780     Subjekt, dessen Zustand sie beobachten und speichern und dessen
781     Zustand konsistent ist. Sie implementieren eine
782     Aktualisierungsschnittstelle unter Verwendung der
783     Abfrageschnittstelle des konkreten Subjekts.
784     \footcite{wiki:beobachter}
785   \end{description}
786 }
```

\liEntwurfsBeobachterCode

```

787 \def\liEntwurfsBeobachterCode{
788   \li@EntwurfsCode{beobachter}{Gegenstand}
789   \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
790   \li@EntwurfsCode{beobachter}{Beobachter}
791   \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
792   \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
```

```

793 \li@EntwurfsCode{beobachter}{Klient}
794 }

```

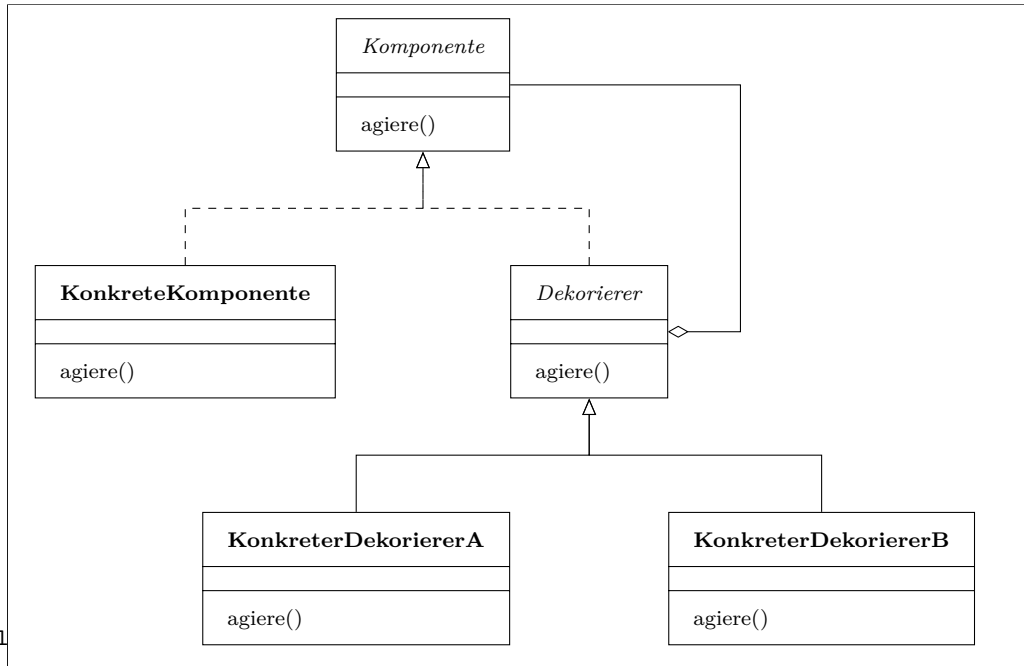
\liEntwurfsBeobachter

```

795 \def\liEntwurfsBeobachter{
796 \liEntwurfsBeobachterUml
797 \liEntwurfsBeobachterAkteure
798 \liEntwurfsBeobachterCode
799 }

```

## 2.10.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

800 \def\liEntwurfsDekoriererUml{
801 \begin{tikzpicture}
802 \umlclass[type=abstract]{Komponente}{-}{agiere()}
803 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
804 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
805
806 \umlVHVreal{KonkreteKomponente}{Komponente}
807 \umlVHVreal{Dekorierer}{Komponente}
808
809 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
810 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
811
812 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
813 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
814
815 \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
816 \footcite{wiki:dekorierer}
817 \end{tikzpicture}
818 }

```

\liEntwurfsDekoriererCode

```

819 \def\liEntwurfsDekoriererCode{
820 \li@EntwurfsCode{dekorierer}{Komponente}
821 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
822 \li@EntwurfsCode{dekorierer}{Dekorierer}
823 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
824 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
825 \li@EntwurfsCode{dekorierer}{Klient}
826 }

```

\liEntwurfsDekorierer

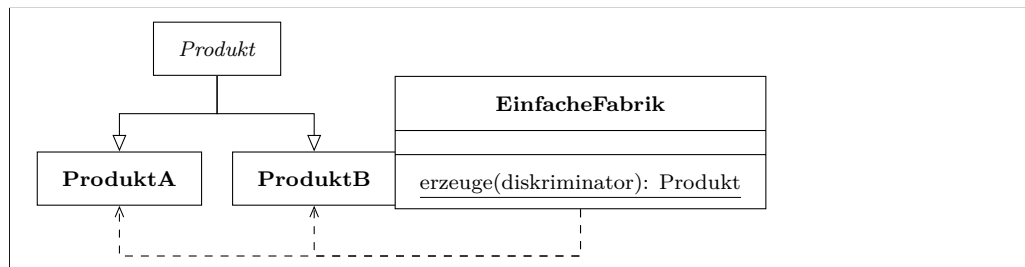
```

827 \def\liEntwurfsDekorierer{
828   \liEntwurfsDekoriererUml
829   \liEntwurfsDekoriererAkteure
830   \liEntwurfsDekoriererCode
831 }

```

## 2.10.7 Einfache Fabrik (Simple Factory)

\liEntwurfsEinfacheFabrikUml Quelle: <https://refactoring.guru/design-patterns/factory-comparison>



```

832 \def\liEntwurfsEinfacheFabrikUml{
833   \begin{tikzpicture}
834     \umlsimpleclass[type=abstract]{Produkt}
835     \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
836     \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
837     \umlVHVinherit{Produkt}{ProduktA}
838     \umlVHVinherit{Produkt}{ProduktB}
839     \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
840     }{
841       \umlstatic{erzeuge(diskriminator): Produkt}\\
842     }
843     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
844     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
845   \end{tikzpicture}
846 }

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

**EinfacheFabrik** Eine Klasse mit einer Erzeugungsmethode, die über eine größere Bedingung verschiedene Objekt instanziiert.

**Produkt** Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.

**KonkretesProdukt** Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.

```

847 \def\liEntwurfsEinfacheFabrikAkteure{
848   \begin{description}
849     \item[EinfacheFabrik]
850
851     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
852     Bedingung verschiedene Objekt instanziiert.
853
854     \item[Produkt]
855
856     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
857
858     \item[KonkretesProdukt]
859
860     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
861   \end{description}
862 }

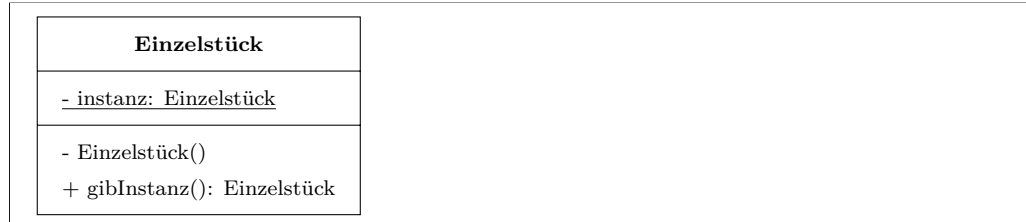
```

\liEntwurfsEinfacheFabrik

```
863 \def\liEntwurfsEinfacheFabrik{
864   \liEntwurfsEinfacheFabrikUml
865   \liEntwurfsEinfacheFabrikAkteure
866 }
```

### 2.10.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```
867 \def\liEntwurfsEinzelstueckUml{
868   \begin{tikzpicture}
869     \umlclass{Einzelstück}{
870       \umlstatic{- instanz: Einzelstück}\\
871     }{
872       - Einzelstück()\\
873       + gibInstanz(): Einzelstück
874     }
875   \end{tikzpicture}
876 }
```

iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

**Einzelstück (Singleton)** stellt eine statische Methode bereit, mit deren Hilfe die Klienten nur auf eine einzige Instanz der Klasse zugreifen können.

```
877 \def\liEntwurfsEinzelstueckAkteure{
878   \begin{description}
879     \item[Einzelstück (Singleton)]
880
881     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
882     nur auf eine einzige Instanz der Klasse zugreifen können.
883   \end{description}
884 }
```

\liEntwurfsEinzelstueckCode

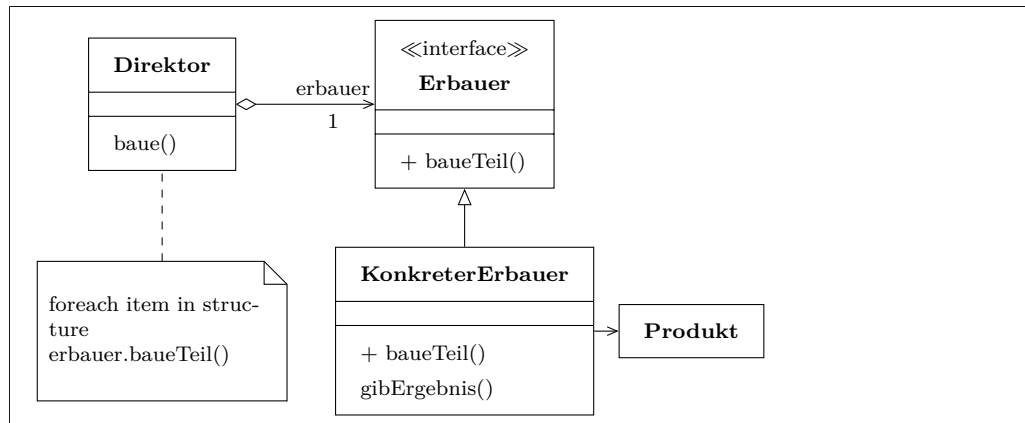
```
885 \def\liEntwurfsEinzelstueckCode{
886   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
887 }
```

\liEntwurfsEinzelstueck

```
888 \def\liEntwurfsEinzelstueck{
889   \liEntwurfsEinzelstueckUml
890   \liEntwurfsEinzelstueckAkteure
891   \liEntwurfsEinzelstueckCode
892 }
```

### 2.10.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

893 \def\liEntwurfsErbauerUml{
894   \begin{tikzpicture}
895     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
896     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
897     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
898       + baueTeil()\
899       gibErgebnis()}
900     \umlclass[x=0,y=3]{Direktor}{}{baue()}
901
902     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
903     \umluniassoc{KonkreterErbauer}{Produkt}
904     \umlinherit{KonkreterErbauer}{Erbauer}
905
906     \umlnote[x=0,y=0,width=3cm]{Direktor}{
907       foreach item in structure\
908       erbauer.baueTeil()
909     }
910   \end{tikzpicture}
911   \footcite{wiki:erbauer}
912 }

```

\liEntwurfsErbauerAkteure Quelle: deutsche Wikipedia

**Erbauer** Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der Teile eines komplexen Objektes.

**KonkreterErbauer** Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch Implementierung der Schnittstelle. Außerdem definiert und verwaltet er die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine Schnittstelle zum Auslesen des Produkts.

**Direktor** Der Direktor konstruiert ein komplexes Objekt unter Verwendung der Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom Klienten.

**Produkt** Das Produkt repräsentiert das zu konstruierende komplexe Objekt.

```

913 \def\liEntwurfsErbauerAkteure{
914   \begin{description}
915     \item[Erbauer]
916
917     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
918     Teile eines komplexen Objektes.
919
920     \item[KonkreterErbauer]
921
922     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
923     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er

```

```

924     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
925     Schnittstelle zum Auslesen des Produkts.
926
927     \item[Direktor]
928
929     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
930     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
931     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
932     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
933     Klienten.
934
935     \item[Produkt]
936
937     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
938     \footcite{wiki:erbauer}
939 \end{description}
940 }

```

\liEntwurfsErbauer

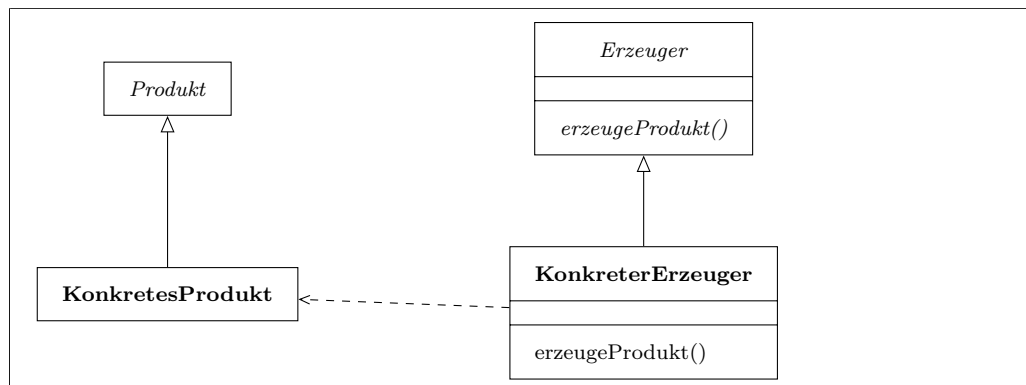
```

941 \def\liEntwurfsErbauer{
942   \liEntwurfsErbauerUml
943   \liEntwurfsErbauerAkteure
944 }

```

## 2.10.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

945 \def\liEntwurfsFabrikmethodeUml{
946   \begin{tikzpicture}
947     \umlsimpleclass[type=abstract]{Produkt}
948     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
949     \umlinherit{KonkretesProdukt}{Produkt}
950
951     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
952       \textit{erzeugeProdukt()}\}
953     }
954     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
955       erzeugeProdukt()
956     }
957     \umlinherit{KonkreterErzeuger}{Erzeuger}
958
959     \umldep{KonkreterErzeuger}{KonkretesProdukt}
960   \end{tikzpicture}
961 }

```

EntwurfsFabrikmethodeAkteure Quelle: deutsche Wikipedia



**Produkt** Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das zu erzeugende Produkt.

**KonkretesProdukt** KonkretesProdukt implementiert die Produkt-Schnittstelle.

**Erzeuger** Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt zu erzeugen und kann eine Default-Implementierung beinhalten.

**KonkreterErzeuger** KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den Konstruktor einer konkreten Produkt-Klasse aufruft).

```

962 \def\liEntwurfsFabrikmethodeAkteure{
963   \begin{description}
964     \item[Produkt]
965
966     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
967     zu erzeugende Produkt.
968
969     \item[KonkretesProdukt]
970
971     KonkretesProdukt implementiert die Produkt-Schnittstelle.
972
973     \item[Erzeuger]
974
975     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
976     zu erzeugen und kann eine Default-Implementierung beinhalten.
977
978     \item[KonkreterErzeuger]
979
980     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
981     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
982     Konstruktor einer konkreten Produkt-Klasse aufruft).
983
984     \footcite{wiki:fabrikmethode}
985   \end{description}
986 }

```

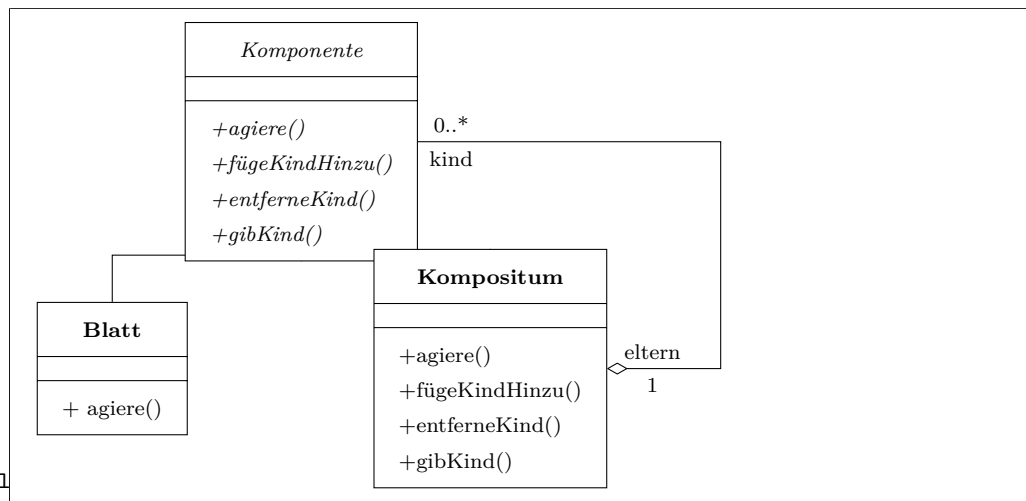
\liEntwurfsFabrikmethode

```

987 \def\liEntwurfsFabrikmethode{
988   \liEntwurfsFabrikmethodeUml
989   \liEntwurfsFabrikmethodeAkteure
990 }

```

### 2.10.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

991 \def\liEntwurfsKompositumUml{
992   \begin{tikzpicture}
993     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
994       \textit{+agiere()}\
995       \textit{+fügeKindHinzu()}\
996       \textit{+entferneKind()}\
997       \textit{+gibKind()}
998     }
999     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1000    \umlclass[x=5]{Kompositum}{-}{
1001      +agiere()\
1002      +fügeKindHinzu()\
1003      +entferneKind()\
1004      +gibKind()
1005    }
1006
1007    \umlVHVinherit{Kompositum}{Komponente}
1008    \umlVHVinherit{Blatt}{Komponente}
1009    \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1010  \end{tikzpicture}
1011 }

```

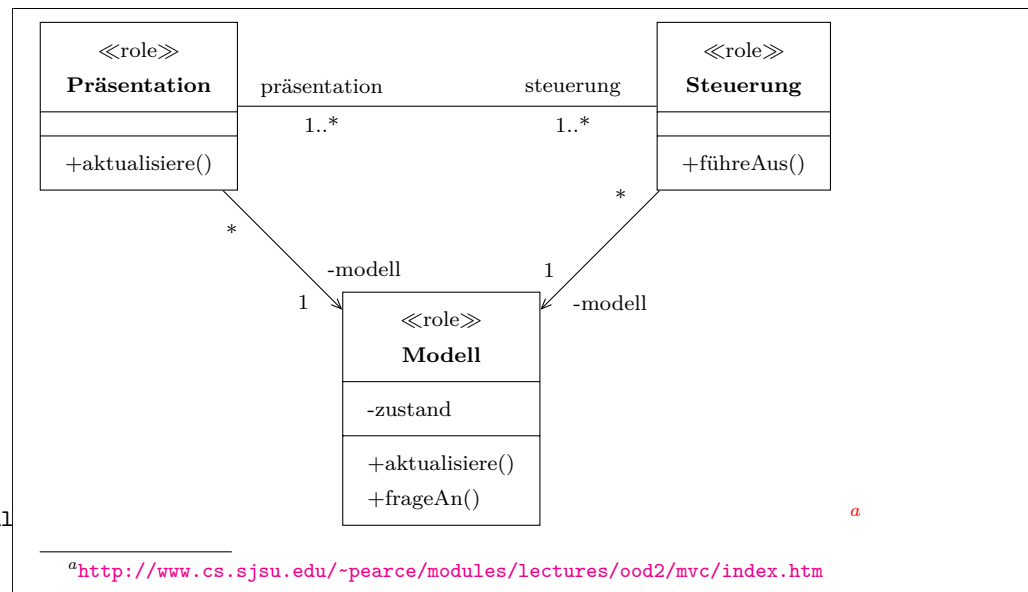
\liEntwurfsFabrikmethode

```

1012 \def\liEntwurfsKompositum{
1013   \liEntwurfsKompositumUml
1014   \liEntwurfsKompositumAkteure
1015 }

```

## 2.10.12 Modell-Präsentation-Steuerung (Model-View-Controller)



```

1016 \def\liEntwurfsModellPraesentationSteuerungUml{
1017   \begin{tikzpicture}
1018     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1019     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1020     \umlclass[y=-4cm,type=role]{Modell}{-}{
1021       -zustand
1022     }{
1023       +aktualisiere()\
1024       +frageAn()
1025     }
1026
1027     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1028     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1029 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1030 \end{tikzpicture}
1031 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1032 }

```

ModellPraesentationSteuerung

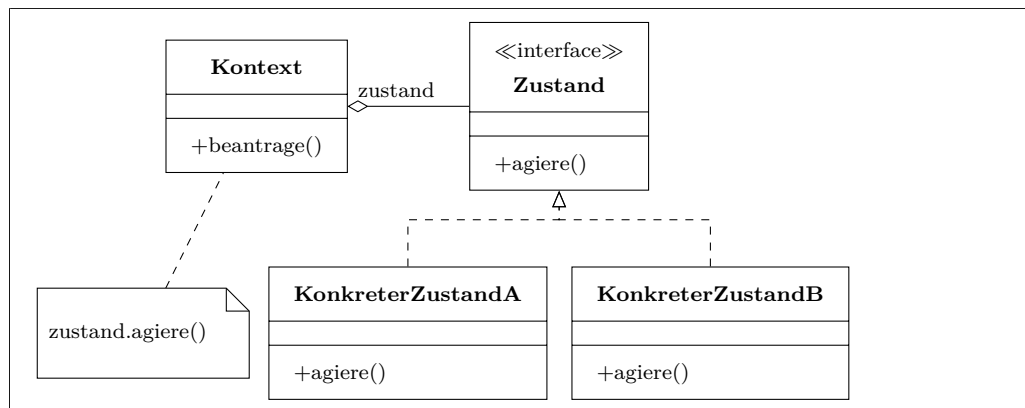
```

1033 \def\liEntwurfs{
1034 \liEntwurfsModellPraesentationSteuerungUml
1035 \liEntwurfsModellPraesentationSteuerungAkteure
1036 }

```

### 2.10.13 Zustand (State)

\liEntwurfsZustandUml Quelle: Englische Wikipedia, so ähnlich wie in GoF



```

1037 \def\liEntwurfsZustandUml{
1038 \begin{tikzpicture}
1039 \umlcass[x=-1,y=3]{Kontext}{+beantrage()}
1040 \umlcass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1041 \umlcass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1042 \umlcass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1043
1044 \umlVHVreal{KonkreterZustandA}{Zustand}
1045 \umlVHVreal{KonkreterZustandB}{Zustand}
1046
1047 \umlagg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1048
1049 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1050 \end{tikzpicture}
1051 }

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

**Kontext (Context)** definiert die clientseitige Schnittstelle und verwaltet die separaten Zustandsklassen.

**State (Zustand)** definiert eine einheitliche Schnittstelle aller Zustandsobjekte und implementiert gegebenenfalls ein Standardverhalten.

**KontreterZustand (ConcreteState)** implementiert das Verhalten, das mit dem Zustand des Kontextobjektes verbunden ist.

```

1052 \def\liEntwurfsZustandAkteure{
1053 \begin{description}
1054 \item[Kontext (Context)]
1055
1056 definiert die clientseitige Schnittstelle und verwaltet die separaten
1057 Zustandsklassen.
1058

```

```

1059 \item[State (Zustand)]
1060
1061 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1062 implementiert gegebenenfalls ein Standardverhalten.
1063
1064 \item[KontreterZustand (ConcreteState)]
1065
1066 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1067 verbunden ist.
1068 \end{description}
1069 }

\liEntwurfsZustand
1070 \def\liEntwurfsZustand{
1071 \liEntwurfsZustandUml
1072 \liEntwurfsZustandAkteure
1073 }

1074

```

## 2.11 er.sty

```
1075 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1076 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1077 ER-Diagrammen]

1078 \RequirePackage{tikz-er2}
1079 \usetikzlibrary{positioning}

\begin{tikzpicture}[er2,scale=0.7,transform shape]
% Person
\node[entity] (Person) {Person};
\node[attribute,right=1cm of Person] {\key{E-Mail}} edge (Person);
\node[multi attribute,above left=1cm of Person] {Vornamen} edge (Person);
\node[attribute,left=1cm of Person] {Nachnamen} edge (Person);
\node[attribute,above right=1cm of Person] {Geburtsdatum} edge (Person);

% Kunde
\node[entity,below left=1cm of Person] (Kunde) {Kunde};

% Händler
\node[entity,below right=1cm of Person] (Händler) {Händler};

\node[specialization,below=0.2cm of Person]{is-a}
edge (Kunde) edge (Händler) edge (Person);

% Transaktion
\node[relationship,below=2cm of Person] (Transaktion) {Transaktion}
edge node[auto]{1} (Kunde)
edge node[auto]{1} (Händler);
\node[attribute,below=1cm of Transaktion] {Preis} edge (Transaktion);
\node[attribute,left=1cm of Transaktion,text width=2cm] {Abschlussdatum} edge (Transaktion);
\node[attribute,right=1cm of Transaktion] {Lieferadresse} edge (Transaktion);

% Zahlungsmittel
\node[entity,below=4cm of Händler] (Zahlungsmittel) {Zahlungsmittel}
edge node[auto]{1} (Transaktion);
\node[attribute,right=1cm of Zahlungsmittel] {Inhaber} edge (Zahlungsmittel);

% Bankverbindung
\node[entity,below left=1cm and 0cm of Zahlungsmittel] (Bankverbindung) {Bankverbindung};
\node[attribute,below left=1cm of Bankverbindung] {\key{IBAN}} edge (Bankverbindung);

% Kreditkarte
\node[entity,below right=3cm and -2cm of Zahlungsmittel]
(Kreditkarte) {Kreditkarte};

\node[attribute,below left=1cm of Kreditkarte]
{\key{Nummer}} edge (Kreditkarte);

\node[attribute,below right=1cm of Kreditkarte,text width=2cm]
{Ablaufdatum} edge (Kreditkarte);

\node[attribute,right=1cm of Kreditkarte]
{Anbieter} edge (Kreditkarte);

\node[generalization,below=0.8cm of Zahlungsmittel]{is-a}
edge (Zahlungsmittel) edge (Bankverbindung) edge (Kreditkarte);

% Produkt
\node[entity,below=4cm of Kunde] (Produkt) {Produkt}
edge node[auto]{1} (Transaktion);
\node[attribute,left=1cm of Produkt] {\key{Bezeichnung}} edge (Produkt);
\node[attribute,below left=1cm of Produkt,text width=2cm] {Beschreibungstext} edge (Produkt);
\node[attribute,above left=1cm of Produkt] {Bewertung} edge (Produkt);
```

```

\end{tikzpicture}

1080 \RequirePackage{soul}
1081 \RequirePackage{fontawesome}

Let-Abkürzungen
\let\ a=\liErMpAttribute
\let\ d=\liErDatenbankName
\let\ e=\liErMpEntity
\let\ r=\liErMpRelationship

1082 \ExplSyntaxOn

\liErEntity
1083 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
1084 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
1085 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\ e=\liErMpEntity
1086 \def\liErMpEntity#1{
1087   \liErEntity{#1}
1088   \marginpar{
1089     \liErEntity{\tiny\faSquareO{}}~E:~#1}
1090   }
1091 }
□

\liErMpRelationship Let-Abkürzung: \let\ r=\liErMpRelationship
1092 \def\liErMpRelationship#1{
1093   \liErRelationship{#1}
1094   \marginpar{
1095     \liErRelationship{\tiny\faGg{}}~R:~#1}
1096   }
1097 }

\liErMpAttribute Let-Abkürzung: \let\ a=\liErMpAttribute
1098 \def\liErMpAttribute#1{
1099   \liErAttribute{#1}
1100   \marginpar{
1101     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1102   }
1103 }

\liErDatenbankName Let-Abkürzung: \let\ d=\liErDatenbankName
datenbank name
1104 \def\liErDatenbankName#1{
1105   {
1106     \footnotesize\texttt{( #1 )}
1107   }
1108 }

1109 \ExplSyntaxOff
1110

```

## 2.12 formale-sprachen.sty

```

1111 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1112 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1113 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1114 \directlua{
1115   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1116 }

1117 \RequirePackage{hyperref}

1118 \liLadePakete{mathe,typographie}

\liMenge   $\liMenge{a, b, c}$:  $\{a, b, c\}$ 
Let-Abkürzung: \let\m=\liMenge

1119 \def\liMengeOhneMathe#1{\{ #1 \}}
1120 \def\liMenge#1{%
1121   \ifmode%
1122     \liMengeOhneMathe{#1}%
1123   \else%
1124     $\liMengeOhneMathe{#1}$%
1125   \fi%
1126 }

\liEpsilon \liEpsilon:  $\varepsilon$ 
Let-Abkürzung: \let\e=\liEpsilon

1127 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1128 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1129 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1130 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

\liZustandsmenge \liZustandsmenge{z1, z2}:  $\{z_1, z_2\}$ 

1131 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1132 \let\liZustandsmenge=\liPotenzmenge

\liUeberfuehrungsFunktion \liUeberfuehrungsFunktion{z0, a}:  $\delta(z_0, a)$ 
Let-Abkürzung: \let\d=\liUeberfuehrungsFunktion

1133 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1134 \def\liUeberfuehrungsFunktion#1{
1135   \ifmode
1136     \liUeberfuehrungsFunktionOhneMathe{#1}
1137   \else
1138     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1139   \fi
1140 }

\liAlphabet \liAlphabet{a, b} ( $\Sigma = \{a, b\}$ ):  $\Sigma = \{a, b\}$ 

1141 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

\liBandAlphabet \liBandAlphabet{\liTuringLeerzeichen}:  $\Gamma = \Sigma \cup \{\square\}$ 

1142 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1143 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1144 \def\liZustandsBuchstabeGross{Z}

```

\liZustandsmengeNr

```

1145 \def\zustandsnamens@liste#1#2{
1146   $
1147   \{
1148     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1149   \}
1150   $
1151 }
1152 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

```

\liZustandsmengeNrGross

```

1153 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

```

\liZustandsname

```

\liZustandsname{1}: $z_1$
1154 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

```

\liZustandsnameGross

```

\liZustandsnameGross{1}: $Z_1$
1155 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

```

\liAbleitung

```

\liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1156 \def\liAbleitung#1{\directlua{formale_sprachen.formatiere_ableitung('#1')}}$}

```

liProduktionsRegeln

```

\begin{liProduktionsRegeln}[P_1]
  S -> S A B | EPSILON,
  B A -> A B,
  A A -> a a,
  B B -> b b
\end{liProduktionsRegeln}

1157 \NewDocumentEnvironment { liProduktionsRegeln }
1158 { 0{P} +b }
1159 {
1160   \liGeschweifteKlammern{#1}
1161   {
1162     \begin{align*}
1163       \directlua{formale_sprachen.produktions_regeln('#2')}
1164       \end{align*}
1165     }{-0.2cm}{-1.5cm}
1166   } {}

```

\liProduktionen

```

\liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1167 \def\liProduktionen#1{
1168   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1169 }

```

\liZustandsnameTiefgestellt

Automatisch tiefgestellte Nummerierung \z1  
**Let-Abkürzung:** \let\z=\liZustandsnameTiefgestellt

```

1170 \def\liZustandsnameTiefgestellt#1{
1171   \ifmmode
1172     \liZustandsBuchstabe\sb{#1}
1173   \else
1174     $\liZustandsBuchstabe\sb{#1}$
1175   \fi
1176 }

```

```

1177 \ExplSyntaxOn

```

\liAusdruck

```

\liAusdruck[L_2]{a_1,a_2,\dots,a_n}{n \in N}:  $L_2 = \{a_1, a_2, \dots, a_n \mid n \in N\}$ 
  Ohne „=: \liAusdruck[] {x}{y}:  $\{x \mid y\}$ 
  Regulärer Ausdruck zum Konvertieren:
  \$(.*) += +\\{ *(.*) ( *\\, *)?\\ ( *\\, *)?(.*) *\\}\\$
  \liAusdruck[$1]{$2}{$5}

```



```

1178 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1179   $
1180   \tl_if_empty:nTF {#1} {} {#1 =}
1181   \{
1182     \, #2 \,
1183     |
1184     \, #3 \,
1185   \}$
1186 }
1187 \ExplSyntaxOff

\liFlaci   Link zur flaci.com Website: \liFlaci{Grxk1oczg}:
           Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte Automaten, Compiler und Interpreter)
           Ein Projekt der Hochschule Zittau/Görlitz und der Pädagogischen Hochschule Schwyz: flaci.com/Grxk1oczg

1188 \def\liFlaci#1{%
1189   \par
1190   {%
1191     \scriptsize
1192     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1193     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1194     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1195     \href{https://flaci.com/#1}{flaci.com/#1}%
1196   }%
1197   \par
1198 }

\liGrammatik \liGrammatik[\langle grammatik-name \rangle]{\langle variablen=V,alphabet=\Sigma,produktionen=P,start=S \rangle}
              \liGrammatik{variablen={}, alphabet={}}

              • \liGrammatik{}:  $G = (V, \Sigma, P, S)$ 

              • \liGrammatik[G_1]{}:  $G_1 = (V, \Sigma, P, S)$ 

              • \liGrammatik{variablen={S,A,B}}:  $G = (\{S, A, B\}, \Sigma, P, S)$ 

              • \liGrammatik{alphabet={a,b}}:  $G = (V, \{a, b\}, P, S)$ 

              • \liGrammatik{start=X}:  $G = (V, \Sigma, P, X)$ 

1199 \ExplSyntaxOn
1200 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1201   \tl_set:Nn \l_variablen_tl {V}
1202   \tl_set:Nn \l_alphabet_tl {\Sigma}
1203   \tl_set:Nn \l_produktionen_tl {P}
1204   \tl_set:Nn \l_start_tl {S}
1205
1206   \keys_define:nn { grammatik } {
1207     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1208     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1209     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1210     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1211   }
1212
1213   \keys_set:nn { grammatik } { #2 }
1214
1215   $#1 = (
1216     \l_variablen_tl,
1217     \l_alphabet_tl,
1218     \l_produktionen_tl,
1219     \l_start_tl
1220   )$
1221 }
1222 \ExplSyntaxOff

1223

```

## 2.13 formatierung.sty

```
1224 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1225 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

### 2.13.1 Schriftarten / Typographie

```
1226 \RequirePackage{mathpazo}
1227 \RequirePackage[no-math]{fontspec}
1228 \setmainfont{texgyrepagella}
1229 \setsansfont{QTAncientOlive}
1230 \RequirePackage{sectsty}
1231 \allsectionsfont{\fontspec{QTAncientOlive}}
```

### 2.13.2 Farben

```
1232 \RequirePackage{xcolor}
1233 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

### 2.13.3 Überschriften

```
1234 \RequirePackage{titlesec}
1235 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1236 \titlespacing{\chapter}{0pt}{0pt}{*1}
1237 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1238 \setcounter{secnumdepth}{0}
```

### 2.13.4 Listen

```
1239 \RequirePackage{paralist}
1240 \renewcommand\labelitemi{-}
1241 \renewcommand\labelitemii{-}
1242 \renewcommand\labelitemiii{-}
1243 \renewcommand\labelitemiv{-}
1244 % Counter: enumi enumii enumiii enumiv
1245 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1246 \renewcommand{\labelenumi}{(\alph{enumi})}
1247 \renewcommand{\labelenumii}{(\roman{enumii})}
```

### 2.13.5 Kasten

```
1248 \RequirePackage{mdframed}
1249 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1250 \NewDocumentEnvironment { liKasten }{ } {
1251   \begin{mdframed}[backgroundcolor=white!0]
1252 } {
1253   \end{mdframed}
1254 }
```

### 2.13.6 Header

```
1255 \RequirePackage{fancyhdr}
1256 \fancyhead[L,C,R]{}
1257 \fancyfoot[L]{}
1258 \fancyfoot[C]{}
1259 \fancyfoot[R]{\thepage}
1260 \pagestyle{fancy}
1261 \renewcommand{\headrulewidth}{0pt}
1262 \renewcommand{\footrulewidth}{0pt}
1263
```

## 2.14 gantt.sty

```

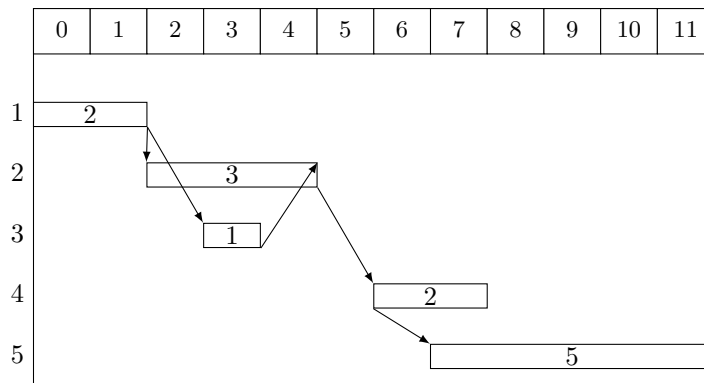
1264 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1265 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\begin{ganttchart}[x unit=0.75cm, y unit chart=0.8cm]{0}{11}
\gantttitlelist{0,...,11}{1} \\
\ganttbar[name=1]{1}{0}{1} \\
\ganttbar[name=2]{2}{2}{4} \\
\ganttbar[name=3]{3}{3}{3} \\
\ganttbar[name=4]{4}{6}{7} \\
\ganttbar[name=5]{5}{7}{11}

\node at (1) {2};
\node at (2) {3};
\node at (3) {1};
\node at (4) {2};
\node at (5) {5};

\ganttlink[link type=f-f]{3}{2}
\ganttlink[link type=f-s]{1}{2}
\ganttlink[link type=f-s]{1}{3}
\ganttlink[link type=f-s]{2}{4}
\ganttlink[link type=s-s]{4}{5}
\end{ganttchart}

```



```

1266 \RequirePackage{tikz-uml}
1267 \RequirePackage{pgfgantt}
1268 \setganttlinklabel{f-s}{}
1269 \setganttlinklabel{s-s}{}
1270 \setganttlinklabel{f-f}{}
1271 \setganttlinklabel{s-f}{}

1272

```

## 2.15 grafik.sty

```
1273 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1274 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1275 TikZ, um Grafiken zeichnen zu können.]
1276 \RequirePackage{tikz}
1277
```

## 2.16 graph.sty

```

1278 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1279 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1280 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1281 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

\[
\begin{blockarray}{ccccc}
& a & b & c & d & e \\
\begin{block}{c(ccccc)}
a & 0 & 1 & 0 & 4 & 0 \\
b & 0 & 0 & 0 & 1 & 3 \\
c & 0 & 0 & 0 & 1 & 0 \\
d & 4 & 1 & 1 & 0 & 2 \\
e & 0 & 0 & 0 & 0 & 0
\end{block}
\end{blockarray}
\]

```

$$\begin{array}{c}
 a \\ b \\ c \\ d \\ e
 \end{array}
 \begin{array}{ccccc}
 a & b & c & d & e \\
 \left( \begin{array}{ccccc}
 0 & 1 & 0 & 4 & 0 \\
 0 & 0 & 0 & 1 & 3 \\
 0 & 0 & 0 & 1 & 0 \\
 4 & 1 & 1 & 0 & 2 \\
 0 & 0 & 0 & 0 & 0
 \end{array} \right)
 \end{array}$$

```

1282 \RequirePackage{blkarray}
1283 \usetikzlibrary{arrows.meta}

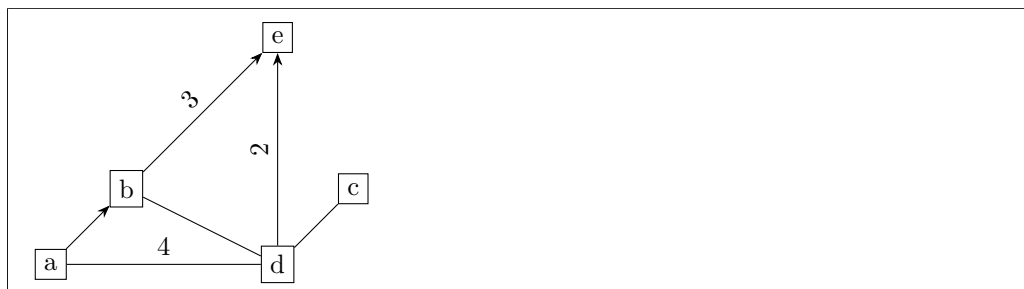
```

```

\begin{tikzpicture}[li graph]
\node (a) at (0,0) {a};
\node (b) at (1,1) {b};
\node (c) at (4,1) {c};
\node (d) at (3,0) {d};
\node (e) at (3,3) {e};

\path[->] (a) edge (b);
\path (b) edge (d);
\path[->] (b) edge node {3} (e);
\path (c) edge (d);
\path (d) edge node {4} (a);
\path[->] (d) edge node {2} (e);
\end{tikzpicture}

```



```

1284 \tikzset{
1285   li graph/.style={
1286     every node/.style={
1287       rectangle,
1288       draw,
1289     },
1290     every edge/.style={
1291       >={Stealth[black]},
1292       draw,
1293     },
1294     every edge/.append style={
1295       every node/.style={
1296         sloped,
1297         auto,
1298       }
1299     }
1300   },
1301   li markierung/.style={
1302     ultra thick,
1303   }
1304 }

```

**liGraphenFormat** Dummy-Umgebung, zum Parsen durch Java gedacht.

```

\begin{liGraphenFormat}
a: 0 0
b: 1 1
c: 4 1
d: 3 0
e: 2 2
a -> b
b -- d
b -> e: 3
c -- d
d -> e: 2
d -- a: 4
\end{liGraphenFormat}

```

```

1305 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1306

```

## 2.17 hanoi.sty

1307 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 1308 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen  
 1309 von Hanoi-Grafiken]

Quelle: <https://kleinco.de/latex-snippets/7/tikz-towers-of-hanoi-illustration-for-lat>

```

1310 \RequirePackage{tikz}
1311 \RequirePackage{xcolor}

\liHanoi \liHanoi{anzahl-scheiben}{gewicht-scheibe/turm-nr,gewicht-scheibe/turm-nr},
z. B. : \liHanoi{4}{4/1,3/1,2/3,1/2}

1312 \def\li@mset #1[#2]=#3{%
1313 \expandafter\xdef\csname #1#2\endcsname{#3}
1314 }
1315 \def\li@mget #1[#2]{%
1316 \csname #1#2\endcsname
1317 }
1318 \def\li@minc #1[#2] += #3{%
1319 \pgfmathparse{\li@mget #1[#2] + #3}%
1320 \li@mset #1[#2] = \pgfmathresult
1321 }
1322
1323 \def\liHanoi#1#2{
1324   \edef\li@numdiscs{#1}
1325   \def\li@sequence{#2}
1326   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1327     % init colors
1328     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1329       \li@mset col[\j]=\c;
1330     }
1331     % draw poles and init pole counters
1332     \foreach \j in {1,2,3}{
1333       \li@mset pos[\j]=0
1334       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1335     }
1336     % draw base
1337     \draw (.5,-.5) -- +(3,0);
1338     % draw discs
1339     \foreach[count=\k] \i/\j in \li@sequence{
1340       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1341       \li@minc pos[\j] += {.5}
1342     }
1343   \end{tikzpicture}
1344 }
```

## 2.18 komplexitaetstheorie.sty

```

1345 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1346 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1347 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1348 Polynomialzeitreduktion.]

```

### Let-Abkürzungen

```

\let\n=\liProblemName
\let\r=\liPolynomiellReduzierbar
\let\b=\liProblemBeschreibung

```

```

1349 \liLadePakete{mathe}

```

Für das Makro `\liProblemBeschreibung{ }{ }` benötigt.

```

1350 \RequirePackage{mdframed}

```

```

\liStrich $L, \liStrich{L}$:  $L, L'$ 

```

```

1351 \def\liStrich#1{#1'\prime}

```

`\liProblemName` Zu Setzen von Problemnamen wie zum Beispiel SAT, COL, VERTEX COVER

**Let-Abkürzung:** `\let\n=\liProblemName`

```

\liProblemName: SAT VERTEX COVER

```

```

1352 \def\liProblemName#1{\texttt{\textsc{#1}}}

```

`\liProblemBeschreibung` Zu setzen von Problem-Beschreibungen:

```

\liProblemBeschreibung
{}
{}
{}

```

### CLIQUE

**Gegeben:** Ein ungerichteter Graph  $G = (V, E)$ , eine Zahl  $k \in \mathcal{N}$

**Frage:** Gibt es eine Menge  $S \subseteq V$  mit  $|S| = k$ , sodass für alle Knoten  $u \neq v \in V$  gilt, dass  $\{u, v\}$  eine Kante in  $E$  ist?

**Let-Abkürzung:** `\let\b=\liProblemBeschreibung`

```

1353 \def\liProblemBeschreibung#1#2#3{
1354   \begin{mdframed}[
1355     userdefinedwidth=9cm,
1356     align=center,
1357     backgroundcolor=white!0,
1358   ]
1359     \centerline{\large\liProblemName{#1}}
1360
1361     \medskip
1362
1363     \begin{description}
1364       \item[Gegeben:] #2
1365       \item[Frage:] #3
1366     \end{description}
1367   \end{mdframed}
1368 }

```



```

\liPolynomiellReduzierbar Let-Abkürzung: \let\r=\liPolynomiellReduzierbar

1369 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1370 \begin{displaymath}
1371 \liProblemName{#1}
1372 \preceq_{#2}
1373 \liProblemName{#3}
1374 \end{displaymath}
1375 }

\liProblemVertexCover

1376 \def\liProblemClique{%
1377 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1378 Mindestgröße  $n$  in einem gegebenen Graphen.
1379 \footcite[Seite 76]{theo:fs:4}
1380 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1381 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1382 \footcite{wiki:cliquenproblem}
1383 }

\liProblemVertexCover

1384 \def\liProblemVertexCover{%
1385 %
1386 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1387 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1388 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1389 \footcite{wiki:knotenueberdeckung}
1390
1391 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1392 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1393 mindestens einem Knoten aus  $U$  verbunden ist.
1394 \footcite[Seite 78]{theo:fs:4}%
1395 }

\liProblemSubsetSum Kein Karp-21-Problem

1396 \def\liProblemSubsetSum{%
1397 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1398 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1399 \footcite{wiki:teilsommenproblem}
1400 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1401 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1402 größer als eine gegebene obere Schranke  $c$  ist.
1403 \footcite[Seite 74]{theo:fs:4}
1404 }

\liProblemSubsetSum Kein Karp-21-Problem

1405 \def\liProblemSat{%
1406 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1407 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1408 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1409 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1410 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1411 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1412 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1413 aufgestellt werden.
1414 \footcite[Seite 71]{theo:fs:4}
1415 }

1416

```

## 2.19 kontrollflussgraph.sty

```
1417 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1418 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1419 \RequirePackage{tikz}
1420 \usetikzlibrary{positioning}
1421 \tikzset{
1422   li kontrollfluss/.style={
1423     knoten/.style={
1424       circle,
1425       draw
1426     },
1427     usebox/.style={
1428       draw,
1429       rectangle,
1430       font=\scriptsize,
1431       anchor=west,
1432       align=left,
1433     },
1434     bedingung/.style={
1435       midway,
1436       draw=none,
1437       font=\scriptsize
1438     },
1439     knotenbeschriftung/.style={
1440       draw,
1441       rectangle,
1442       midway,
1443       font=\scriptsize
1444     },
1445     wahr/.style={
1446       thick
1447     },
1448     falsch/.style={
1449       dashed
1450     },
1451     every node/.style={
1452       circle,
1453       draw,
1454     },
1455     every edge/.append style={
1456       every node/.style={
1457         draw=none,
1458         bedingung,
1459       }
1460     },
1461     every path/.style={
1462       draw,
1463       ->,
1464     },
1465     every pin/.style={
1466       draw,
1467       dotted,
1468       rectangle,
1469       pin position=right
1470     },
1471     every pin edge/.style={
1472       dotted,
1473       arrows=-,
1474     }
1475   }
1476 }
```

liKontrollflussgraph

```

1477 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1478   \begin{tikzpicture}[
1479     li kontrollfluss,
1480     #1
1481   ]
1482 } {
1483   \end{tikzpicture}
1484 }

\liAnweisung
1485 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1486 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1487 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1488 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1489 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1490 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1491 \ExplSyntaxOn
1492 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1493 {
1494   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1495   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1496   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1497 }
1498 \ExplSyntaxOff

1499

```

## 2.20 kopf-fusszeilen.sty

```
1500 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1501 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1502 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1503 \ExplSyntaxOn

1504 \fancyhead{}
1505 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1506 \fancyhead[L0,RE]{\scriptsize\today}
1507 \fancyfoot{}
1508 \fancyfoot[LE,R0]{\thepage}
1509 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1510 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1511 \renewcommand{\headrulewidth}{0.4pt}
1512 \renewcommand{\footrulewidth}{0.4pt}

1513 \ExplSyntaxOff

1514
```

## 2.21 literatur-dummy.sty

```
1515 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1516 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1517 \def\literatur{}

\footcite

1518 % \RequirePackage[stable,multiple]{footmisc}
1519 \NewDocumentCommand{ \footcite } { o m }{}

1520
```

## 2.22 literatur.sty

```
1521 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1522 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1523 \RequirePackage{csquotes}
1524 \RequirePackage[
1525   bibencoding=utf8,
1526   citestyle=authortitle,
1527   backend=biber,
1528 ]{biblatex}
1529 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1530 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1531 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1532 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1533 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1534 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1535 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1536 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1537 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1538 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1539 % To allow footnotes in the heading
1540 \RequirePackage[stable,multiple]{footmisc}

\literatur

1541 \def\literatur{\printbibliography[heading=subbibliography]}

1542
```

## 2.23 makros.sty

```
1543 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1544 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1545 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1546 anderen Paket passen]
1547 \RequirePackage{hyperref}
1548 \RequirePackage{graphicx}

    Für die Umgebung liQuellen benötigt.
1549 \RequirePackage{paralist}

\inhaltsverzeichnis
1550 \def\inhaltsverzeichnis {
1551     \begin{mdframed}
1552         \begin{group}
1553             \let\clearpage\relax
1554             \tableofcontents
1555         \end{group}
1556     \end{mdframed}
1557 }

\memph \mephm (\marginpar and \emph)
1558 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1559 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1560 \newcommand{\liPseudoUeberschrift}[1]{
1561     \bigskip
1562     \noindent
1563     \textsf{\textbf{#1}}
1564     \noindent
1565 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1566 \newcommand{\liBeschriftung}[1]{
1567     \par
1568     \noindent
1569     \medskip
1570     \textbf{#1}:
1571     \medskip
1572     \noindent
1573 }

\hinweis
1574 \def\hinweis#1{{\footnotesize#1}}

liProjektSprache \begin{liProjektSprache}{NameProjektSprache} \end{liProjektSprache}: Zum Einbetten von projekteigenen Minisprachen/DSLs (Domain-specific language) (z. B. RelationenSchema). Der Inhalt der Umgebung wird nicht von TeX kompiliert, sondern von dem Java-Kommandozeilen-Programm didaktik.java verarbeitet.
1575 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1576 \NewDocumentEnvironment { liEinbettung }{ o +b } {#2} {}

liAntwort Zum Setzen von Antworten. Sie werden mit einem Kasten umgeben. Könnten eventuell ausgeblendet werden, wenn man nur die Lösung sehen will.
1577 \RequirePackage{xparse}
1578 \ExplSyntaxOn
```

```

1579 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1580 {
1581   \str_case:nn {#1} {
1582     {standard} {
1583       \def\beschriftung{}
1584       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1585     }
1586     {richtig} {
1587       \def\beschriftung{richtig}
1588       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1589     }
1590     {falsch} {
1591       \def\beschriftung{falsch}
1592       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1593     }
1594     {muster} {
1595       \def\beschriftung{Musterlösung}
1596       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1597     }
1598   }
1599   \ifx\beschriftung\empty\else
1600   \noindent
1601   \textbf{\beschriftung{:}}
1602   \fi
1603   \begin{mdframed}
1604 }
1605 {\end{mdframed}}

```

**liAdditum** Zusätzliches Material bei Aufgaben, das zum Lösen der Aufgaben nicht unbedingt nötig ist, z. b. Hintergrundinformation, Test-Dateien, komplette Code-Dateien.

```

1606 \NewDocumentEnvironment{ liAdditum } { o }
1607 {
1608   \begin{mdframed}[backgroundcolor=yellow!5]
1609   \IfNoValueTF {#1}
1610   { \liPseudoUeberschrift{Additum} }
1611   { \liPseudoUeberschrift{Additum:~#1} }
1612 }
1613 {\end{mdframed}}

```

**liExkurs** \begin{liExkurs}[Linear rekursiv]  
Die häufigste Rekursionsform ist die lineare Rekursion, bei der in jedem Fall der rekursiven Definition höchstens ein rekursiver Aufruf vorkommen darf.  
\end{liExkurs}

#### **Exkurs: Linear rekursiv**

Die häufigste Rekursionsform ist die lineare Rekursion, bei der in jedem Fall der rekursiven Definition höchstens ein rekursiver Aufruf vorkommen darf.

```

1614 \NewDocumentEnvironment{ liExkurs }{o +b}{
1615   \vspace{0.2cm}%
1616   \begin{mdframed}[
1617     backgroundcolor=white,
1618     bottomline=false,
1619     innermargin=1cm,
1620     leftline=true,
1621     linecolor=black,
1622     linewidth=0.1cm,
1623     outermargin=1cm,
1624     rightline=false,
1625     topline=false,
1626   ]

```



```

1627 \footnotesize
1628 \noindent%
1629 \textbf{Exkurs:~#1}\par%
1630 \noindent%
1631 #2
1632 \end{mdframed}
1633 \vspace{0.2cm}
1634 }{}

```

liQuellen <https://tex.stackexchange.com/a/229004>

```

\begin{liQuellen}
\item Quelle 1
\item Quelle 2
\end{liQuellen}

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

1635 \cs_new:Npn \listen@punkt #1 {\item #1}
1636 \NewDocumentEnvironment { liQuellen }{ +b }
1637 {
1638   \seq_clear_new:N \l_quellen
1639   \seq_set_split:Nnn \l_quellen {\item} {#1}
1640   \seq_remove_all:Nn \l_quellen {}
1641   \begin{mdframed}[backgroundcolor=white!0]
1642     \footnotesize
1643     \noindent
1644     \textsf{\textbf{Weiterführende~Literatur:}}
1645     \medskip
1646     \begin{compactitem}
1647       \seq_map_function:NN \l_quellen {\listen@punkt}
1648     \end{compactitem}
1649   \end{mdframed}
1650   %
1651   \makeatletter
1652   \par\@afterindentfalse\@afterheading
1653   \makeatother
1654 } {}

```

liLernkartei

```

1655 \NewDocumentEnvironment { liLernkartei }{ m +b }
1656 {
1657   \begin{mdframed}
1658     \footnotesize
1659     \noindent%
1660     \textbf{Lernkarteikarte:~#1}\par%
1661     \noindent%
1662     #2
1663   \end{mdframed}
1664 } {}

```

liDiagramm \begin{liDiagramm}{beschriftung}\end{liDiagramm}: Zu setzen einer Graphik bzw eines Diagramms.

```

1665 \NewDocumentEnvironment { liDiagramm }{ m +b }
1666 {
1667   \begin{mdframed}[backgroundcolor=white!0]
1668     \small
1669     \noindent%
1670     \textit{#1}:
1671     \begin{center}

```

```

1672 #2
1673 \medskip
1674 \end{center}
1675 \end{mdframed}
1676 } {}
1677 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1678 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1679 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1680 }
1681

\liFussnoteLink \liFussnoteLink[\langle zusätzlicher-text \rangle]{\langle link-text \rangle}{\langle url \rangle} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
1682 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1683 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1684 }

\zB
1685 \def\zB{z.\,B. }

\ZB
1686 \def\ZB{Z.\,B. }

\dh
1687 \def\dh{d.\,h. }

1688

```

## 2.24 master-theorem.sty

1689 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1690 \ProvidesPackage{lehramt-informatik-master-theorem}[2021/04/13]

### Let-Abkürzungen

\let\0=\li0

\let\o=\li0mega

\let\T=\liT

\let\t=\liTheta

\liMasterVariablenDeklaration

{3} % a

{3} % b

{\mathcal{O}(1)} % f(n)

\liMasterFallRechnung

% 1. Fall

{für  $\varepsilon = 4$ : \}

$f(n) = 5n^2 \in \mathcal{O}(n^{\log_2 8 - 4}) = \mathcal{O}(n^{\log_2 4}) = \mathcal{O}(n^2)$

% 2. Fall

$f(n) = 5n^2 \notin \mathcal{T}(n^{\log_2 8}) = \mathcal{T}(n^3)$

% 3. Fall

$f(n) = 5n^2 \notin \mathcal{O}(n^{\log_2 8 + \varepsilon})$

\liMasterWolframLink{T[n]=9T[n/3]\%2B5n^2}

\liMasterVariablenDeklaration

{ } % a

{ } % b

{ } % f(n) ohne \$mathe\$

\liMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

$\rightarrow T(n) \in \mathcal{T}(n^2 \cdot \log n)$

\liMasterWolframLink{T[n]=9T[n/3]\%2B5n^2}

1691 \ExplSyntaxOn

1692 \RequirePackage{amsmath}

\liRundeKlammer

1693 \def\liRundeKlammer#1{

1694 \negthinspace \left( #1 \right)

1695 }

\liTheta \liTheta{n^2}:  $\Theta(n^2)$

1696 \def\liThetaOhneMathe#1{

1697 \Theta \liRundeKlammer{#1}

1698 }

1699 \def\liTheta#1{

1700 \ifmmode

1701 \liThetaOhneMathe{#1}

1702 \else

1703 \$\liThetaOhneMathe{#1}\$

1704 \fi

1705 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1706 \def\liOmegaOhneMathe#1{
1707 \Omega \liRundeKlammer{#1}
1708 }
1709 \def\liOmega#1{
1710 \ifmmode
1711 \liOmegaOhneMathe{#1}
1712 \else
1713 $\liOmegaOhneMathe{#1}$
1714 \fi
1715 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1716 \def\liOOhneMathe#1{
1717 \mathcal{O} \liRundeKlammer{#1}
1718 }
1719 \def\liO#1{
1720 \ifmmode
1721 \liOOhneMathe{#1}
1722 \else
1723 $\liOOhneMathe{#1}$
1724 \fi
1725 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1726 \def\liTOhneMathe#1#2{
1727 \tl_if_blank:nTF {#1}
1728 {}
1729 {#1 \cdot }
1730 T
1731 \liRundeKlammer{\frac{n}{#2}}
1732 }
1733 \def\liT#1#2{
1734 \ifmmode
1735 \liTOhneMathe{#1}{#2}
1736 \else
1737 $\liTOhneMathe{#1}{#2}$
1738 \fi
1739 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1740 \def\liRekursionsGleichung{
1741 $T(n) = \liT{a}{b} + f(n)$
1742 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1743 \def\liBedingungEins{
1744 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1745 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1746 \def\liBedingungZwei{
1747 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1748 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1749 \def\liBedingungDrei{
1750 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1751 }

1752 \ExplSyntaxOff

```

## \liMasterVariablen

```

1753 \def\liMasterVariablen{
1754   \begin{displaymath}
1755     T(n) = \liT{a}{b} + f(n)
1756   \end{displaymath}
1757
1758   \begin{itemize}
1759     \item[$a = $]
1760       Anzahl der Unterprobleme in der Rekursion
1761
1762     \item[$\textstyle{\frac{1}{b}} = $]
1763       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1764       repräsentiert wird
1765
1766     \item[$f(n) = $]
1767       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1768       die Kombination der Teillösungen entstehen
1769   \end{itemize}
1770   \footcite{wiki:master-theorem}
1771   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1772 }

```

## \liMasterFaelle

```

1773 \def\liMasterFaelle{
1774   \begin{description}
1775     \item[1. Fall:]
1776        $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1777
1778       \hfill falls \liBedingungEins
1779       für  $\varepsilon > 0$ 
1780
1781     \item[2. Fall:]
1782        $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1783
1784       \hfill falls \liBedingungZwei
1785
1786     \item[3. Fall:]
1787        $T(n) \in \liTheta{f(n)}$ 
1788
1789       \hfill falls \liBedingungDrei
1790       für  $\varepsilon > 0$ 
1791       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1792       gilt:
1793        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1794     \end{description}
1795 }

```

## \liMasterVariablenDeklaration

```

1796 \def\liMasterVariablenDeklaration#1#2#3{
1797   \begin{description}
1798     \item[Allgemeine Rekursionsgleichung:] \strut
1799
1800     \liRekursionsGleichung
1801
1802     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1803
1804     #1
1805
1806     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1807
1808     um  $\frac{1}{#2}$  also  $b = #2$ 
1809
1810     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut

```

```

1811
1812     $#3$
1813
1814     \item[Ergibt folgende Rekursionsgleichung:] \strut
1815
1816     $T(n) = \liT{\#1}{\#2} + \#3$
1817 \end{description}
1818 }

\liMasterFallRechnung
1819 \def\liMasterFallRechnung#1#2#3{
1820 \begin{description}
1821 \item[1. Fall:] \liBedingungEins:
1822
1823     #1
1824
1825 \item[2. Fall:] \liBedingungZwei:
1826
1827     #2
1828
1829 \item[3. Fall:] \liBedingungDrei:
1830
1831     #3
1832 \end{description}
1833 }

\liMasterExkurs
1834 \def\liMasterExkurs{
1835 \begin{liExkurs}[Master-Theorem]
1836 \liMasterVariablen
1837
1838 \noindent
1839 Dann gilt:
1840
1841 \liMasterFaelle
1842 \end{liExkurs}
1843 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1844 \def\liMasterWolframLink#1{
1845 Berechne die Rekursionsgleichung auf WolframAlpha:
1846 \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
1847 }

1848

```

## 2.25 mathe.sty

```
1849 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1850 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1851
1852 % for example \ltimes \rtimes
1853 %\RequirePackage{amssymb}
1854 \RequirePackage{amsmath}
1855
1856 %%
1857 % \mlq \mrq
1858 %%
1859 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1860 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1861
```

## 2.26 minimierung.sty

```

1862 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1863 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1864 Minimierungsalgorithmus von einem NEA zu einem DEA]

1865 \liLadePakete{typographie}

\let\z=\liZustandsnameTiefgestellt
\let\f=\liFussnote
\let\l=\liLeereZelle
\let\Z=\liZustandsPaar
\let\erklaerung=\liMinimierungErklaerung

\begin{tabular}{|c||c|c|c|c|c|c|c|c|c|}
\hline
\z0 & \l & \l & \l & \l & \l & \l & \l & \l & \l & \l \\
\z1 & & \l & \l & \l & \l & \l & \l & \l & \l & \l \\
\z2 & & & \l & \l & \l & \l & \l & \l & \l & \l \\
\z3 & & & & \l & \l & \l & \l & \l & \l & \l \\
\z4 & & & & & \l & \l & \l & \l & \l & \l \\
\z5 & & & & & & \l & \l & \l & \l & \l \\
\z6 & & & & & & & \l & \l & \l & \l \\
\z7 & & & & & & & & \l & \l & \l \\
\z8 & & & & & & & & & \l & \l \\
& \z0 & \z1 & \z2 & \z3 & \z4 & \z5 & \z6 & \z7 & \z8 & \l \\
\end{tabular}

\liFussnoten

\begin{liUebergangsTabelle}{0}{1}
\Z01 & \Z10 & \Z23 & \\
\Z05 & \Z15 & \Z25 & \f2 \\
\Z15 & \Z05 & \Z35 & \f2 \\
\Z23 & \Z44 & \Z55 & \\
\Z24 & \Z44 & \Z55 & \\
\Z34 & \Z44 & \Z55 & \\
\end{liUebergangsTabelle}

\liFussnote

1866 \def\liFussnote#1{\$x_{#1}\$}

1867 \def\li@fussnote@text#1#2{
1868 \liFussnote{#1}
1869 \quad
1870 {\footnotesize #2}
1871 }

\liFussnoteEinsText

1872 \def\liFussnoteEinsText{
1873 \li@fussnote@text{1}
1874 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1875 }

\liFussnoteZweiText

1876 \def\liFussnoteZweiText{
1877 \li@fussnote@text{2}
1878 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1879 }

\liFussnoteDreiText

1880 \def\liFussnoteDreiText{
1881 \li@fussnote@text{3}

```



```

1882 {In weiteren Iterationen markierte Zustände.}
1883 }

\liFussnoteVierText
1884 \def\liFussnoteVierText{
1885   \li@fussnote@text{4}
1886   {...}
1887 }

\liFussnoten


|       |                                                                      |
|-------|----------------------------------------------------------------------|
| $x_1$ | Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.          |
| $x_2$ | Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt. |
| $x_3$ | In weiteren Iterationen markierte Zustände.                          |
| $x_4$ | ...                                                                  |



1888 \def\liFussnoten{
1889   \bigskip
1890
1891   \noindent
1892   \liFussnoteEinsText
1893
1894   \noindent
1895   \liFussnoteZweiText
1896
1897   \noindent
1898   \liFussnoteDreiText
1899
1900   \noindent
1901   \liFussnoteVierText
1902 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1903 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1904 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1905 \def\liZustandsPaar#1#2{
1906   $(
1907     \liZustandsPaarVariablenName_#1,
1908     \liZustandsPaarVariablenName_#2
1909   )$
1910 }

liUebergangsTabelle
1911 \renewcommand{\arraystretch}{1.4}
1912 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1913   \liPseudoUeberschrift{Übergangstabelle}
1914   \begin{center}
1915     \begin{tabular}{r|l|l}
1916       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
1917     \end{tabular}
1918   \end{center}
1919 }
1920 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1921 \ExplSyntaxOn
1922 \def\liUeberschriftDreiecksTabelle{
1923   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1924 }

```

**\liMinimierungErklaerung** **Let-Abkürzung:** `\let\verklaerung=\liMinimierungErklaerung`

— Der Minimierungs-Algorithmus (auch Table-Filling-Algorithmus genannt) trägt in seinem Verlauf eine Markierung in alle diejenigen Zellen der Tabelle ein, die zueinander nicht äquivalente Zustände bezeichnen. Die Markierung „ $x_n$ “ in einer Tabellenzelle  $(i, j)$  bedeutet dabei, dass das Zustandspaar  $(i, j)$  in der  $k$ -ten Iteration des Algorithmus markiert wurde und die Zustände  $i$  und  $j$  somit zueinander  $(k - 1)$ -äquivalent, aber nicht  $k$ -äquivalent und somit insbesondere nicht äquivalent sind. Bleibt eine Zelle bis zum Ende unmarkiert, sind die entsprechenden Zustände zueinander äquivalent. —————

```

1925 \def\liMinimierungErklaerung{
1926   %\footcite[Seite~19]{koenig}
1927   \liParagraphMitLinien{
1928     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1929     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1930     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1931     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1932      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1933     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1934     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1935     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1936     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1937   }
1938 }
1939 \ExplSyntaxOff

```

1940

## 2.27 normalformen.sty

```

1941 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1942 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
1943 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1944 Attributhülle]

```

Mathe für textit tex etc, typographie für geschweifte Klammer

```

1945 \liLadePakete{mathe,typographie}
1946 \directlua{
1947   helper = require('lehramt-informatik-helfer')
1948   normalformen = require('lehramt-informatik-normalformen')
1949 }

```

### Let-Abkürzungen

```

\let\ah=\liAttributHuelle
\let\ahL=\liLinksReduktion
\let\ahL=\liLinksReduktionInline
\let\ahr=\liRechtsReduktionInline
\let\fa=\liFunktionaleAbhaengigkeit
\let\FA=\liFunktionaleAbhaengigkeiten
\let\m=\liAttributMenge
\let\r=\liRelation
\let\u=\underline

```

```

1950 \def\liTeilen#1{
1951   \directlua{tex.print(normalformen.teilen('#1'))}
1952 }

```

**\liAttributHuelle Let-Abkürzung:** `\let\ah=\liAttributHuelle`  
`\ah{F, \m{A, B}}` AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren  
AttrHülle $((.*)\backslash)$  `\ah{$1}`

```

1953 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
1954 \def\liAttributHuelle#1{
1955   \ifmmode
1956     \liAttributHuelleOhneMathe{#1}
1957   \else
1958     $\liAttributHuelleOhneMathe{#1}$
1959   \fi
1960 }

```

**\liAttributMenge Let-Abkürzung:** `\let\m=\liAttributMenge`

```

1961 \def\liAttributMenge#1{\{ \textit{#1} \}}

```

**liAHuelle**

```

1962 \NewDocumentEnvironment{ liAHuelle } { +b } {
1963   \begin{group}
1964   \footnotesize
1965   \begin{multline*}
1966     #1
1967   \end{multline*}
1968   \end{group}
1969 } { }

```

**\liLinksReduktion** Nur innerhalb von liAHuelle zu verwenden bzw. multline

**Let-Abkürzung:** `\let\ahL=\liLinksReduktion`  
`\ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}`

```

1970 \def\liLinksReduktion#1#2#3{
1971   \shoveleft{
1972     \liAttributHuelleOhneMathe{FA,
1973       \liAttributMenge{#1 \string\ #2}} =
1974   } \\\

```

```

1975 \shoveright{
1976   \liAttributMenge{#3}
1977 } \\\
1978 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahL=\liLinksReduktionInline
    \ahl{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
\ahl{A, E}{E}{A, E, F, B, \textbf{D}}: AttrHülle( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
1979 \def\liLinksReduktionInline#1#2#3{
1980   $\liAttributHuelleOhneMathe{F,
1981     \liAttributMenge{#1 \string\ #2}} =
1982   \liAttributMenge{#3}$
1983 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
1984 \def\liRechtsReduktionInline#1#2#3#4{
1985   $\liAttributHuelleOhneMathe{
1986     F \setminus
1987     \liFunktionaleAbhaengigkeit{#1}
1988     \def\tmp{#2}\ifx\tmp\empty
1989     \else
1990       \cup \liFunktionaleAbhaengigkeit{#2}
1991     \fi
1992     ,
1993     \liAttributMenge{#3}
1994   } =
1995   \liAttributMenge{#4}$
1996 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \rightarrow (.*)\$ \fa{$1 -> $2}
1997 \def\liFunktionaleAbhaengigkeit#1{%
1998   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
1999 }

```

```

\liFunktionaleAbhaengigkeiten FA[F]{
    M -> M;
    M -> N;
    V -> T, P, PN;
    P -> PN;
}

```

$$F = \left\{ \begin{array}{l} \{M\} \rightarrow \{M\}, \\ \{M\} \rightarrow \{N\}, \\ \{V\} \rightarrow \{T, P, PN\}, \\ \{P\} \rightarrow \{PN\}, \end{array} \right\}$$

```

Let-Abkürzung: \let\FA=\liFunktionaleAbhaengigkeiten
    Regulärer Ausdruck zum Konvertieren: \item \$(.*) \rightarrow (.*)\$ \$(.*) \rightarrow
    $1 -> $2;
2000 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2001   \liGeschweifteKlammern
2002   {#1}
2003   {
2004     \begin{align*}
2005     \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}

```

```

2006      \end{align*}
2007    }
2008    {-0.5cm}
2009    {-1.7cm}
2010 }

\liRelation Let-Abkürzung: \let\r=\liRelation
      \r[R3]{\u{A}, B, C}:  $R_3(\underline{A}, B, C)$  Regulärer Ausdruck zum Konvertieren:
      \$(R.*)\((.*)\)\$ \liRelation[1]{2}
2011 \NewDocumentCommand {\liRelation} { O{R} m } {
2012   $\directlua{
2013     local name = helper.konvertiere_tiefgestellt('#1')
2014     tex.print(name)
2015   }$(\textit{\,#2\,})
2016 }

2017

```

## 2.28 petri.sty

2018 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2019 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### Let-Abkürzungen

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2020 \RequirePackage{tikz}

2021 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2022 \RequirePackage{blkarray}

```
\def\TmpA#1{
  \liPetriSetzeSchluessel%
  \pgfkeys{/petri/.cd,#1}%
  \begin{tikzpicture}[li petri]
    \node at (-0.25,-0.25) {};
    \node at (\TmpX,\TmpY) {};

    \begin{scope}[transform canvas={scale=\TmpScale},x=2cm,y=2cm,]
      \node[place,tokens=\TmpPlaceOne,label=$p_1$] at (0,1) (p1) {};
      \node[place,tokens=\TmpPlaceTwo,label=$p_2$] at (2,2) (p2) {};
      \node[place,tokens=\TmpPlaceThree,label=east:$p_3$] at (2,0) (p3) {};

      \node[transition,label=east:$t_1$, \TmpTransitionOne] at (2,1) {}
        edge[pre] (p2)
        edge[post] (p3);
      \node[transition,label=$t_2$, \TmpTransitionTwo] at (1,1.5) {}
        edge[pre] (p1)
        edge[post] (p2);
      \node[transition,label=$t_3$, \TmpTransitionThree] at (1,0.5) {}
        edge[pre] (p3)
        edge[post] (p1);
      \node[transition,label=$t_4$, \TmpTransitionFour] at (1,1) {}
        edge[pre] (p2)
        edge[pre] (p3)
        edge[post] (p1);
    \end{scope}
  \end{tikzpicture}
}
```

\liPetriSetzeSchluessel

```
2023 \def\liPetriSetzeSchluessel{%
2024   \def\TmpTransitionOne{}%
2025   \def\TmpTransitionTwo{}%
2026   \def\TmpTransitionThree{}%
2027   \def\TmpTransitionFour{}%
2028   \def\TmpTransitionFive{}%
2029   \def\TmpTransitionSix{}%
2030   \def\TmpTransitionSeven{}%
2031   \def\TmpTransitionEight{}%
2032   \def\TmpTransitionNine{}%
2033   \def\TmpTransitionTen{}%
2034   \pgfkeys{/petri/.cd,
2035     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2036     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2037     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2038     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
2039     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

2040 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2041 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2042 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2043 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2044 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2045 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2046 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2047 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2048 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2049 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2050 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2051 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2052 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2053 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2054 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2055 scale/.store in=\TmpScale,scale/.default=0.5,
2056 x/.store in=\TmpX,x/.default=5,
2057 y/.store in=\TmpY,y/.default=5,
2058 }%
2059 }

2060 \tikzset{
2061   li petri/.style={
2062     activated/.style={
2063       very thick
2064     },
2065     inhibitor/.style={
2066       {Circle[open,length=2mm,fill=white]}-
2067     }
2068   }
2069 }

```

**\liPetriTransitionsName** **Let-Abkürzung:** \let\t=\liPetriTransitionsName  
 \t\_(\d+)\\$ \t\$1

```

2070 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2071 \def\liPetriTransitionsName#1{
2072   \ifmmode
2073     \liPetriTransitionsNameOhneMathe{#1}
2074   \else
2075     $\liPetriTransitionsNameOhneMathe{#1}$
2076   \fi
2077 }

```

**\liPetriErreichTransition** **Let-Abkürzung:** \let\t=\liPetriErreichTransition

```

2078 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2079   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2080 }

```

**\liPetriErreichKnotenDrei** **Let-Abkürzung:** \let\k=\liPetriErreichKnotenDrei

```

2081 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

**\liPetriTransPfeile** **Let-Abkürzung:** \let\tp=\liPetriTransPfeile

```

2082 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2083 }

```

## 2.29 potenzmengen-konstruktion.sty

```
2084 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2085 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2086 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2087 \liLadePakete{formale-sprachen}
2088 \ExplSyntaxOn
```

```
\def\z#1{
  \liZustandsMengenSammlungNr{#1}{
    {
      {0} {0}
      {1} {0,1}
      {2} {0,2}
      {3} {0,1,3}
      {4} {0,2,3}
      {5} {0,3}
    }
  }
}
\let\s=\liZustandsnameGross

\begin{tabular}{l|l|l}
Zustandsmenge & Eingabe $a$ & Eingabe $b$ \\ \hline
\z0 & \z0 & \z1 \\
\z1 & \z2 & \z1 \\
\z2 & \z0 & \z3 \\
\z3 & \z4 & \z3 \\
\z4 & \z5 & \z3 \\
\z5 & \z5 & \z3 \\
\end{tabular}
```

```
\liZustandsMengenSammlung \liZustandsMengenSammlung{nummer}{latex3 str_case:nn}
```

```
\def\z#1{
  \liZustandsMengenSammlung{#1}{
    {
      {0} {z0}
      {1} {z0, z1}
      {2} {z0, z1, z2}
      {3} {z0, z2}
      {4} {z0, z1, z2, z3}
      {5} {z0, z3}
      {6} {z0, z2, z3}
      {7} {z0, z1, z3}
    }
  }
}
```

```
2089 \def\liZustandsMengenSammlung#1#2{
2090   \liZustandsnameGross{#1}
2091   {
2092     \footnotesize
2093     \liPotenzmenge{
2094       \str_case:nn {#1} {#2
2095       }
2096     }
2097 }
```

```
\liZustandsMengenSammlungNr
```

```
2098 \def\liZustandsMengenSammlungNr#1#2{
2099   \liZustandsnameGross{#1}
2100   {
```



```
2101     \footnotesize
2102     \liZustandsmengeNr{
2103         \str_case:nn {#1} #2
2104     }
2105 }
2106 }

2107 \ExplSyntaxOff
2108
```

## 2.30 pseudo.sty

2109 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2110 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen  
 2111 von Pseudo-Code, Hüll-Paket um algorithm2e]

```
\begin{algorithm}[H]
\KwData{$G = (V, E, w)$: ein zusammenhängender, ungerichteter,
kantengewichteter Graph kruskal(G)}
$E' \leftarrow \emptyset$;
$L \leftarrow E$;
Sortiere die Kanten in $L$ aufsteigend nach ihrem Kantengewicht.;
\While{$L \neq \emptyset$}{
  wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;
  entferne die Kante $e$ aus $L$;
  \If{der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält}{
    $E' \leftarrow E' \cup \{e\}$;
  }
}
\KwResult{$M = (V, E')$ ist ein minimaler Spannbaum von $G$.}
\caption{Minimaler Spannbaum nach Kruskal\footcite{wiki:kruskal}}
\end{algorithm}
```

Algorithmus 1: Minimaler Spannbaum nach Kruskal
---

<p><b>Data:</b> <math>G = (V, E, w)</math>: ein zusammenhängender, ungerichteter, kantengewichteter Graph kruskal(<math>G</math>)</p> <p><math>E' \leftarrow \emptyset</math>;  <math>L \leftarrow E</math>;          Sortiere die Kanten in <math>L</math> aufsteigend nach ihrem Kantengewicht.;</p> <p><b>while</b> <math>L \neq \emptyset</math> <b>do</b></p> <div style="margin-left: 20px;"> <p>wähle eine Kante <math>e \in L</math> mit kleinstem Kantengewicht;</p> <p>entferne die Kante <math>e</math> aus <math>L</math>;</p> <p><b>if</b> der Graph <math>(V, E' \cup \{e\})</math> keinen Kreis enthält <b>then</b></p> <div style="margin-left: 20px;"> <p><math>E' \leftarrow E' \cup \{e\}</math>;</p> </div> <p><b>end</b></p> </div> <p><b>end</b></p> <p><b>Result:</b> <math>M = (V, E')</math> ist ein minimaler Spannbaum von <math>G</math>.</p>
---

2112 \RequirePackage[german,boxruled]{algorithm2e}

2113

## 2.31 pumping-lemma.sty

2114 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2115 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2116 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2117 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2118 \def\liPumpingRegulaer{%
2119   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2120   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2121    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2122   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2123
2124   \begin{enumerate}
2125     \item  $|v| \geq 1$ 
2126     (Das Wort  $v$  ist nicht leer.)
2127
2128     \item  $|uv| \leq j$ 
2129     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2130
2131     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
2132     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2133     Sprache  $L$ )
2134   \end{enumerate}
2135
2136   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2137   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2138 }
```

\liPumpingKontextfrei

```
2139 \def\liPumpingKontextfrei{%
2140   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2141   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
2142    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2143
2144   \begin{enumerate}
2145     \item  $|vx| \geq 1$ 
2146     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2147
2148     \item  $|vwx| \leq j$ 
2149     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2150
2151     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy \in L$  (Für jede
2152     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2153     Sprache  $L$ )
2154   \end{enumerate}
2155 }

2156
```

## 2.32 quicksort.sty

```

2157 % https://tex.stackexchange.com/a/142634
2158 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2159 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2160
2161 %-----
2162 % USAGE:
2163 % \QSinitialize{comma, separated, numerical, values}
2164 % \loop
2165 % \QSpivotStep
2166 % \ifnum\value{pivotcount}>0
2167 %   \QSSortStep
2168 % \repeat
2169 %-----
2170
2171 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2172 \RequirePackage{xintfrac, xinttools}
2173
2174 \RequirePackage{tikz}
2175
2176 %-----
2177 % FIRST PART: TikZ styles and macros for the actual drawing
2178 \newcounter{cellcount}% used for coordinates of the node
2179 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2180
2181 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2182 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2183 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2184 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2185 % by police of LaTeX good conduct ? )
2186 \tikzset{ll/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2187          oo/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2188          rr/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2189 % this is the "b" style as used in the image below
2190 %   bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2191 % nicer:
2192 %   bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2193 %   gg/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2194
2195 % NOTE the b style was originally the same as the r(aised) style apart from
2196 % not being raised, but I find it nicer with a somewhat different
2197 % specification. I have not updated the images though.
2198
2199 % How the nodes are drawn depending on whether on the left of the pivot value
2200 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2201
2202 \def\DecoLEFT #1{%
2203   \xintFor* ##1 in {#1} \do
2204     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2205 }
2206
2207 \def\DecoINERT #1{%
2208   \xintFor* ##1 in {#1} \do
2209     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2210 }
2211
2212 \def\DecoRIGHT #1{%
2213   \xintFor* ##1 in {#1} \do
2214     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2215 }
2216
2217 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2218   \xintFor* ##1 in {#1} \do

```

```

2219     {\stepcounter{cellcount}}%
2220     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2221 }
2222
2223 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2224     \xintFor* ##1 in {#1} \do
2225     {\stepcounter{cellcount}}%
2226     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2227 }
2228
2229 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2230     \xintFor* ##1 in {#1} \do
2231     {\stepcounter{cellcount}}%
2232     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2233 }
2234
2235 %-----
2236 % SECOND PART: the actual sorting routines.
2237
2238 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2239 \def\QS@sort@b #1{\ifcase #1
2240     \expandafter\QS@sort@empty
2241     \or\expandafter\QS@sort@single
2242     \else\expandafter\QS@sort@c
2243     \fi
2244 }%
2245 \def\QS@sort@empty #1{}
2246 \def\QS@sort@single #1{\QS@Ir {#1}}
2247
2248 % This step is to pick the last as pivot.
2249 \def\QS@sort@c #1%
2250     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2251
2252 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2253 % The trick with \xintApplyUnbraced is that for example when selecting
2254 % the elements smaller than pivot, if we had been using \xintApply we
2255 % would have had at the minimum an empty brace pair. Thus we use the
2256 % "unbraced" variant, but then the \QS@select@smaller has added in
2257 % anticipation a level of braces.
2258 \def\QS@sort@d #1#2{%
2259     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2260     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2261     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2262 }%
2263 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2264 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2265 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2266
2267 %
2268 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2269 %
2270 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2271 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2272 % latter must handle correctly an empty argument.
2273
2274 %-----
2275 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2276
2277 % This draws all with suitable highlighting for the newly chosen pivots
2278 % (which will be shown raised)

```

```

2279 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2280             \let\QSIr\DecoINERT
2281             \let\QSIrr\DecoINERT
2282             \let\QSRr\DecoRIGHTwithPivot
2283 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2284             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2285             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2286 }
2287
2288 % This sorts and then draws, showing where the pivot chosen in the previous
2289 % step go. Next time they will have become "inert". If pivotcount is still at
2290 % zero on exit from \QSpivotStep, then this is the signal to stop before
2291 % executing \QSsortStep.
2292 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2293             \def\QSRr {\noexpand\QS@sort@a}%
2294             \def\QSIr {\noexpand\QSIrr}%
2295             \let\QSIrr\relax
2296             \edef\QS@list{\QS@list}%
2297             \let\QSLr\relax
2298             \let\QSRr\relax
2299             \let\QSIr\relax
2300             \edef\QS@list{\QS@list}%
2301             \let\QSLr\DecoLEFT
2302             \let\QSIr\DecoINERTwithPivot
2303             \let\QSIrr\DecoINERT
2304             \let\QSRr\DecoRIGHT
2305 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2306             \setcounter{cellcount}{0}%
2307             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2308 }
2309
2310 \def\QSinitialize #1{%
2311     % first, we convert the comma separated values into a list of braced items
2312     % we use an \edef, and anyhow many \edef's will be used later
2313     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2314     \let\QSRr\DecoRIGHT
2315     % The \QSRr marker mutated to draw the last element as
2316     % pivot and the earlier ones with the suitable style.
2317     %
2318     % The list of marked braced items \QS@list is used both for drawing
2319     % (as here) and for doing the exchange of elements during sort.
2320     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2321             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2322 }
2323

```

## 2.33 relationale-algebra.sty

```

2324 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2325 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2326 \RequirePackage{amsmath}
2327 \RequirePackage{amssymb}

```

```

    Zum Zeichnen von Operatorenbäumen verwenden wir TikZ
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

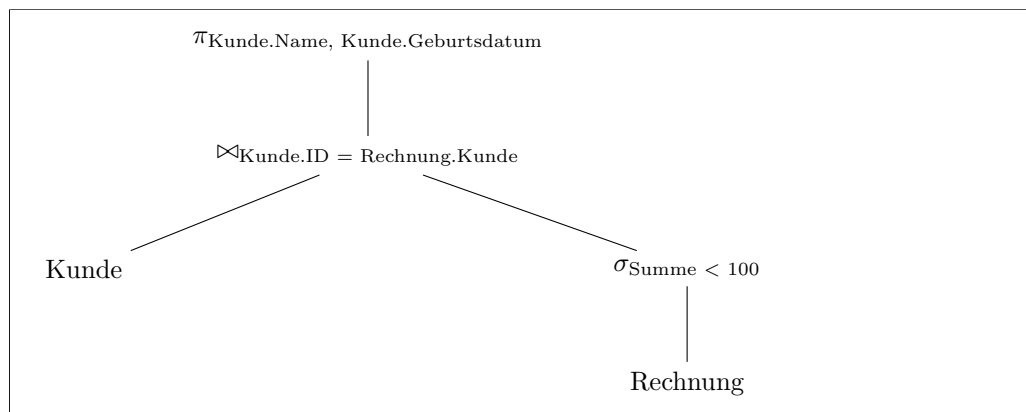
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \node[below left=of theta join]
    {Kunde}
    edge(theta join);

  \node[below right=of theta join]
    (sigma rechnung) {\sigma_{\text{Summe < 100}}};
    edge (theta join);

  \node[below=of sigma rechnung]
    {Rechnung}
    edge(sigma rechnung);
\end{tikzpicture}

```



```

2328 \RequirePackage{tikz}
2329 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2330 \def\o@join{\setbox0=\hbox{\bowtie}}%
2331 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2332 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2333 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2334 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2335 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2336

```

## 2.34 rmodell.sty

```

2337 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2338 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2339 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2340 Datenbanken.]
2341 \RequirePackage{soul}

```

### Let-Abkürzungen

```

\let\a=\liAttribut
\let\f=\liFremd
\let\p=\liPrimaer
\let\r=\liRelationMenge

```

**\liPrimaer** **\liPrimaer{text}**: Unterstreichen für den Primärschlüssel

```
2342 \def\liPrimaer#1{\ul{#1}}
```

**\liFremd** **\liFremd{text}**: Überstreichen für den Fremdschlüssel

```
2343 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

**liRmodell** **\begin{liRmodell}** **\end{liRmodell}**: Kleinere Schrift und Schreibmaschinenschrift.

```

2344 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2345 \ExplSyntaxOn
2346 \NewDocumentEnvironment { liRmodell }
2347 { +b }
2348 {
2349   \medskip
2350   {
2351     \linespread{2}
2352     \setlength{\parindent}{0pt}
2353     \li@Rmodell@Schrift#1
2354   }
2355   \medskip
2356 } {}
2357 \ExplSyntaxOff

```

**\liRelationMenge** **Let-Abkürzung:** **\let\r=\liRelationMenge**

**\liRelationMenge{name}{attribut, attribut}**: Umhüllen der Attribute mit geschweiften und dann eckigen Klammern.

```

2358 \def\liRelationMenge#1#2{
2359 \noindent
2360 #1 : \{ [ #2 ] \}
2361 \par
2362 }

```

**\liAttribut** **Let-Abkürzung:** **\let\a=\liAttribut**

**\liAttribut{text}**: Gleiche Schrift wie Umgebung **liRmodell**

```
2363 \def\liAttribut#1{{\li@Rmodell@Schrift#1}}
```

**liRelationenSchemaFormat** Dummy-Umgebung, zum Parsen durch Java gedacht.

```

\begin{liRelationenSchemaFormat}
Springer(Startnummer*, Nachname, Vorname, Geburtsdatum, Körpergröße)
Sprung(SID*, Beschreibung, Schwierigkeit)
springt(SID[Sprung], Startnummer[Springer], Durchgang)
\end{liRelationenSchemaFormat}

```

```
2364 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2365
```



## 2.35 sortieren.sty

```
2366 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2367 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2368 für händisches Sortieren bzw. einen Schreibtischlauf setzen]
```

```
\tikz[
  rectangle split parts=5,
]{
  \node[li sortierung zahlenreihe] (reihe) {\nodepart{one} 2 \nodepart{two} 1 \nodepart{three} 3 \nodepart{four} 4 \nodepart{five} 5}
  \liSortierPfeil{one}{two}
  \liSortierPfeil{two}{three}
  \liSortierMarkierung{two split south}{three split north}
  \liSortierPfeilUnten{three}{one}
}
```



```
2369 \RequirePackage{tikz}
2370 \usetikzlibrary{shapes.multipart,positioning}
```

`\liVertauschen` `\liVertauschen{1 2 >4 <3 5}`: Setze ein Schaubild mit Hilfe von TikZ. < und > werden dazu verwendet, um den Vertauschprozess zu visualisieren.

```
2371 \def\liVertauschen#1{
2372   \directlua{
2373     local sortieren = require('lehramt-informatik-sortieren')
2374     sortieren('#1')
2375   }
2376 }
```

`\liSortierPfeil`

```
2377 \def\liSortierPfeil#1#2{
2378   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2379 }
```

`\liSortierPfeilUnten`

```
2380 \def\liSortierPfeilUnten#1#2{
2381   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2382 }
```

`\liSortierMarkierung`

```
2383 \def\liSortierMarkierung#1#2{\node[
2384   draw,
2385   very thick,
2386   fit=(reihe.#1) (reihe.#2),
2387   inner sep=0pt
2388 ] {}
2389 }

2390 \tikzset{
2391   li sortierung zahlenreihe/.style={
2392     draw,
2393     thin,
2394     font=\large,
2395     rectangle split horizontal,
2396     rectangle split,
2397   }
2398 }
```

```

2399 % https://tex.stackexchange.com/a/140895
2400 \RequirePackage{forest,xstring}
2401 \usetikzlibrary{calc}
2402
2403 \makeatletter
2404 \pgfmathdeclarefunction{strrepeat}{2}{%
2405   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2406   \let\pgfmathresult\pgfutil@empty
2407   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2408     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2409     \advance\pgfmath@count-1\relax
2410   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2411 \makeatother
2412
2413 \def\myNodes{}
2414
2415 \ExplSyntaxOn
2416 \newcommand*\sortList[1]{%
2417   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2418 \ExplSyntaxOff
2419
2420 \forestset{
2421   sort/.code={%
2422     \pgfmathparse{level()>\forestSortLevel}%
2423     \ifnum\pgfmathresult=0
2424       \StrSubstitute{\forestov{content}}{ }{,}[\myList]%
2425       \sortList\myList
2426       \StrSubstitute{\myList}{,}{ }[\myList]%
2427       \pgfmathparse{strrepeat("1",level())}%
2428       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2429         (m\forestov{name}) {\myList}}%
2430       \pgfmathparse{level()=\forestSortLevel}%
2431       \ifnum\pgfmathresult=1
2432         \forestOget{\forestov{@first}}{name}\forestFirst
2433         \forestOget{\forestov{@last}}{name}\forestLast
2434         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2435           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2436       \fi
2437       \ifnum\forestov{@parent}=0\else
2438         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2439       \fi
2440       \gappto\myNodes{;}%
2441     \fi}}
2442
2443 \forestset{sort level/.code=%
2444   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2445   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2446

```

## 2.36 spalten.sty

```
2447 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2448 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2449 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2450 realisiert werden kann.]
2451 \RequirePackage{multicol}
```

```
\liSpaltenUmbruch \liSpaltenUmbruch: Spezieller Spaltenumbruch, der den Inhalt mit Hilfe von \vfill\strut
nach oben schiebt.
```

```
2452 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2453
```

## 2.37 struktogramm.sty

```
2454 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2455 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2456 Paket struktex zum Setzen von Struktogrammen]
2457 \RequirePackage{struktex}
2458
```

## 2.38 syntax.sty

```
2459 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2460 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2461 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2462 \RequirePackage{xparse}
```

Um das Paket `minted` benutzen zu können, muss `pygmentize` installiert sein. Außerdem müssen die TeX-Dateien mit der Kommandozeilen-Option `--shell-escape` kompiliert werden.

### Let-Abkürzungen

```
\let\j=\liJavaCode
\let\s=\liSqlCode
```

```
2463 \ExplSyntaxOn
2464 \directlua{
2465   syntax = require('lehramt-informatik-syntax')
2466   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2467   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2468   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2469   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2470   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2471   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2472 }
2473 \RequirePackage{hyperref}
2474 \RequirePackage{minted}
2475 % pygmentize -L styles
2476 \usemintedstyle{colorful}
2477 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2478 %\AfterEndEnvironment{minted}{\end{mdframed}}
2479 %\setminted{breaklines=true,linenos}
2480 \setminted{
2481   breaklines=true,
2482   linenos,
2483   fontsize=\footnotesize,
2484 }
```

`\liJavaCode` Im Zeilenfluss einen kurzen Java-Code-Ausschnitt setzen.

**Let-Abkürzung:** `\let\j=\liJavaCode`

```
2485 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

`\liLatexCode` Im Zeilenfluss einen kurzen L<sup>A</sup>T<sub>E</sub>X-Code-Ausschnitt setzen.

```
2486 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2487 \def\li@GithubLink#1#2{
2488   \begin{flushright}
2489     \tiny
2490     Code-Beispiel~auf~Github~ansehen:~
2491     \href{#1}{\nolinkurl{#2}}
2492   \end{flushright}
2493 }
```

`\liJavaDatei` Eine komplette Java-Datei einbinden, die Verzeichnis `./Code/src/main/java/org/bschlangaul` liegt.

```
2494 \NewDocumentCommand{\liJavaDatei}{O{firstline=3} m }{
2495   \inputminted[#1]{java}{
2496     \directlua{
2497       syntax.drucke_absoluten_pfad('#2', false)
2498     }
2499   }
2500   \li@GithubLink
```

```

2501     {\directlua{syntax.drucke_github_url('#2', false)}}
2502     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2503 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2504 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2505     \inputminted[#1]{java}{
2506         \directlua{
2507             syntax.drucke_absoluten_pfad('#2', true)
2508         }
2509     }
2510     \li@GithubLink
2511     {\directlua{syntax.drucke_github_url('#2', true)}}
2512     {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2513 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2514 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2515     \inputminted[#1]{java}{
2516         \directlua{
2517             syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2518         }
2519     }
2520
2521     \li@GithubLink
2522     {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2523     {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2524 }

\liAssemblerCode
2525 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2526 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2527     \inputminted{asm}{#1}
2528 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2529 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2530     \inputminted{componentpascal}{#1}
2531 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2532 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2533 \NewDocumentCommand{\liHaskellDatei}{ m }{
2534     \inputminted{haskell}{#1}
2535 }

2536 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2537 \def\liSqlCode#1{\mintinline{sql}|#1|}
2538

```

## 2.39 syntaxbaum.sty

```
2539 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2540 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2541 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2542 \RequirePackage{tikz-qtrees}
2543
2544 \tikzset{li parsetree/.style={
2545     every internal node/.style={
2546         draw,circle
2547     },
2548     every leaf node/.style={
2549         draw,rectangle
2550     },
2551 }
2552 }
2553
```

## 2.40 synthese-algorithmus.sty

```
2554 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2555 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2556 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2557 Relation in die 3. Normalform]

2558 \liLadePakete{normalformen,mathe,typographie}
2559 \ExplSyntaxOn
```

### Let-Abkürzungen

```
\let\schrittE=\liSyntheseUeberErklaerung
```

```
\let\schrittE=\liSyntheseUeberErklaerung
\begin{enumerate}
\item \schrittE{1}

\begin{enumerate}
\item \schrittE{1-1}
\item \schrittE{1-2}
\item \schrittE{1-3}
\item \schrittE{1-4}
\end{enumerate}

\item \schrittE{2}
\item \schrittE{3}
\item \schrittE{4}
\end{enumerate}
```

### 1. Kanonische Überdeckung

— Die kanonische Überdeckung - also die kleinst mögliche noch äquivalente Menge von funktionalen Abhängigkeiten kann in vier Schritten erreicht werden. —

#### (a) Linksreduktion

— Führe für jede funktionale Abhängigkeit  $\alpha \rightarrow \beta \in F$  die Linksreduktion durch, überprüfe also für alle  $A \in \alpha$ , ob  $A$  überflüssig ist, d. h. ob  $\beta \subseteq \text{AttrHülle}(F, \alpha - A)$ .

#### (b) Rechtsreduktion

— Führe für jede (verbliebene) funktionale Abhängigkeit  $\alpha \rightarrow \beta$  die Rechtsreduktion durch, überprüfe also für alle  $B \in \beta$ , ob  $B \in \text{AttrHülle}(F - (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta - B)), \alpha)$  gilt. In diesem Fall ist  $B$  auf der rechten Seite überflüssig und kann eliminiert werden, d. h.  $\alpha \rightarrow \beta$  wird durch  $\alpha \rightarrow (\beta - B)$  ersetzt. —

#### (c) Löschen leerer Klauseln

— Entferne die funktionalen Abhängigkeiten der Form  $\alpha \rightarrow \emptyset$ , die im 2. Schritt möglicherweise entstanden sind. —

#### (d) Vereinigung

— Fasse mittels der Vereinigungsregel funktionale Abhängigkeiten der Form  $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$ , so dass  $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$  verbleibt. —

### 2. Relationsschemata formen

— Erzeuge für jede funktionale Abhängigkeit  $\alpha \rightarrow \beta \in F_c$  ein Relationenschema  $\mathcal{R}_\alpha := \alpha \cup \beta$ . —

### 3. Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata  $\mathcal{R}_\alpha$  einen Schlüsselkandidaten von  $\mathcal{R}$  bezüglich  $F_c$  enthält, sind wir fertig, sonst wähle einen Schlüsselkandidaten  $\mathcal{K} \subseteq \mathcal{R}$  aus und definiere folgendes zusätzliche Schema:  $\mathcal{R}_\mathcal{K} := \mathcal{K}$  und  $\mathcal{F}_\mathcal{K} := \emptyset$  —



#### 4. Entfernung überflüssiger Teilschemata

— *Eliminiere diejenigen Schemata  $R_\alpha$ , die in einem anderen Relationenschema  $R_{\alpha'}$  enthalten sind, d. h.  $R_\alpha \subseteq R_{\alpha'}$ .*

`\liSyntheseUeberschrift` **Let-Abkürzung:** `\let\schritt=\liSyntheseUeberschrift`

```
2560 \def\liSyntheseUeberschrift#1{
2561   {
2562     \bfseries
2563     \sffamily
2564     \str_case:nn {#1} {
2565       {1} {Kanonische-Überdeckung}
2566       {1-1} {Linksreduktion}
2567       {1-2} {Rechtsreduktion}
2568       {1-3} {Löschen-leerer-Klauseln}
2569       {1-4} {Vereinigung}
2570       {2} {Relationsschemata-formen}
2571       {3} {Schlüssel-hinzufügen}
2572       {4} {Entfernung-überflüssiger-Teilschemata}
2573     }
2574   }
2575 }
```

`\liSyntheseErklaerung` **Let-Abkürzung:** `\let\erklaerung=\liSyntheseErklaerung`

```
2576 \def\li@synthese@erklaerung@texte#1{
2577   \str_case:nn {#1} {
2578     {1} {
2579       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2580       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2581       Schritten-erreicht-werden.
2582     }
2583     {1-1} {
2584       Führe-für-jede-funktionale-Anhängigkeit-
2585       $\alpha\rightarrow\beta$-in-F$-die-Linksreduktion-durch,-
2586       überprüfe-also-für-alle-
2587       $A$-in-$\alpha$,~ob-$A$-überflüssig-ist,~d.h.-ob-
2588       $\beta\subseteqq\liAttributHuelle{F,\alpha\---A}$.
2589     }
2590     {1-2} {
2591       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2592       alle-$B$-in-$\beta$,~ob-$B$-in-$\liAttributHuelle{F---(\alpha\rightarrow\beta)\cup(\alpha\rightarrow\beta\---B)}$,~
2593       $\alpha$-gilt.-In-diesem-Fall-ist-B-auf-der-rechten-Seite-
2594       überflüssig-und-kann-eliminiert-werden,~dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow(\beta\---B)$-
2595       ersetzt.
2596     }
2597     {1-3} {
2598       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2599       entstanden-sind.
2600     }
2601     {1-4} {
2602       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2603       der-Form-$\alpha\rightarrow\beta\sb{1},\dots,\alpha\rightarrow\beta\sb{n}$,~so-dass-$\alpha\rightarrow\beta\sb{1}\cup\dots\cup\beta\sb{n}$-verbleibt.
2604     }
2605     {2} {
2606       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-in-F$\sb{c}$-ein-Relationenschema-$\mathcal{R}\sb{\alpha}$-
2607     }
2608   }
```

```

2615         :=~\alpha~\cup~\beta$.
2616     }
2617     {3} {
2618         Falls-eines-der-in-Schritt-2.-erzeugten-Schemata- $R_{\alpha}$ -
2619         einen-Schlüsselkandidaten-von- $R$ -bezüglich- $F_c$ -
2620         enthält,~sind-wir-fertig,~sonst-wähle-einen-Schlüsselkandidaten-
2621          $K \subseteq R$ -aus-und-definiere-folgendes-
2622         zusätzliche-Schema:~ $R_{\alpha} \setminus K := \emptyset$ -
2623         und- $F_{\alpha} \setminus K := \emptyset$ 
2624     }
2625     {4} {
2626         Eliminiere-diejenigen-Schemata- $R_{\alpha}$ ,~die-in-einem-
2627         anderen-Relationenschema- $R_{\alpha'}$ -enthalten-sind,~d.h.~
2628          $R_{\alpha} \subseteq R_{\alpha'}$ .
2629     }
2630 }
2631 }
2632 \def\liSyntheseErklaerung#1{
2633 {
2634     \itshape
2635     \footnotesize
2636     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2637 }
2638 }

```

\liSyntheseUeberErklaerung **Let-Abkürzung:** \let\schrittE=\liSyntheseUeberErklaerung

```

2639 \def\liSyntheseUeberErklaerung#1{
2640     \liSyntheseUeberschrift{#1}\par
2641     \liSyntheseErklaerung{#1}
2642 }

2643 \ExplSyntaxOff
2644

```

## 2.41 tabelle.sty

2645 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2646 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2647 \RequirePackage{tabularx}

2648

## 2.42 typographie.sty

```
2649 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2650 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2651 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2652 formatierung.sty definiert.]
```

```
2653 \ExplSyntaxOn
```

Mit dem Packet wasysym gab es Unverträglichkeiten, deshalb verwenden wir fontawesome.

```
2654 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2655 \let\liErledigt=\faCheckSquareO
```

```
\liNichtsZuTun \liNichtsZuTun: ∅ Nichts zu tun
```

```
2656 \def\liNichtsZuTun{\emptyset~Nichts~zu~tun}
```

```
\liParagraphMitLinien \liParagraphMitLinien{Lorem ipsum...}:
```

— Lorem ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit —

```
2657 \def\liParagraphMitLinien#1{
```

```
2658 \noindent
```

```
2659 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2660 \enspace
```

```
2661 #1
```

```
2662 \enspace
```

```
2663 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2664 \par
```

```
2665 \medskip
```

```
2666 }
```

```
\liGeschweifteKlammern Große geschweifte Klammer mit Istgleich-Zeichen.
```

$$\text{Variable} = \left\{ \begin{array}{c} \text{Inhalt} \end{array} \right\}$$

```
2667 \def\liGeschweifteKlammern#1#2#3#4{
```

```
2668 \par
```

```
2669 \noindent
```

```
2670 #1 \, $\= \Bigl\{ $\
```

```
2671 \vspace{#3}
```

```
2672 #2
```

```
2673 \vspace{#4}
```

```
2674 \begin{flushright} $\Bigr\} $\end{flushright}
```

```
2675 }
```

```
2676 \ExplSyntaxOff
```

```
2677
```

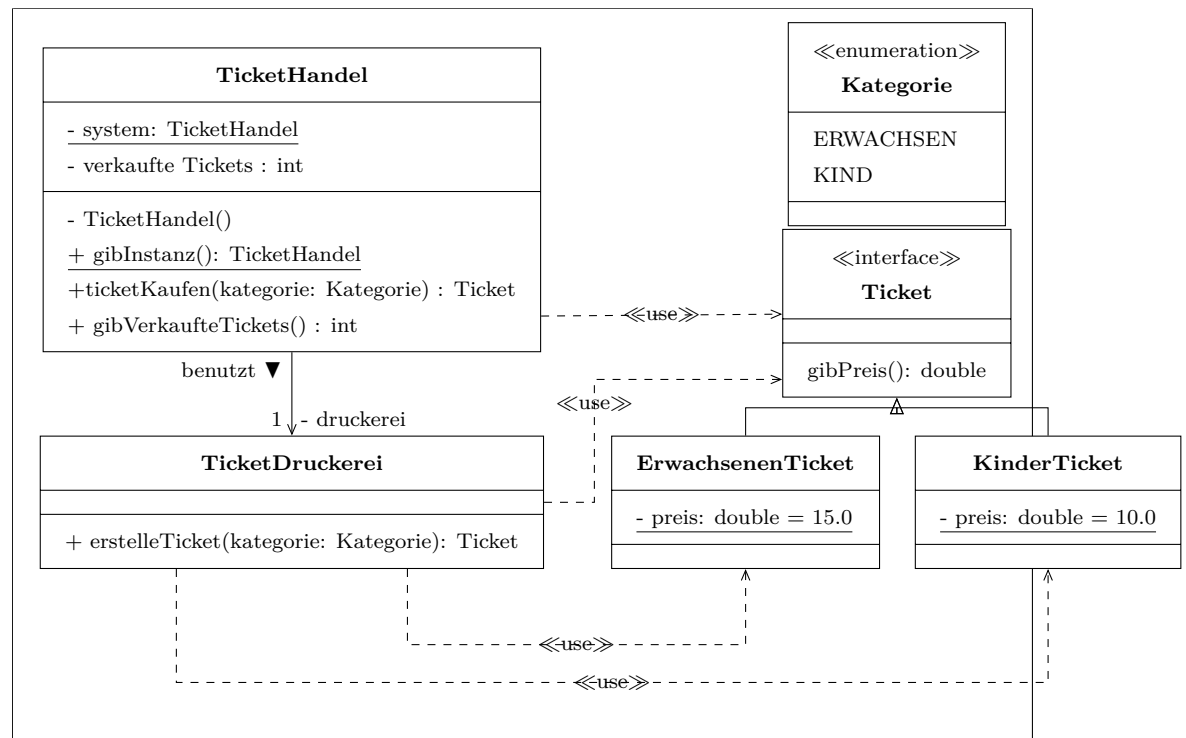
## 2.43 uml.sty

```

2678 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2679 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2680 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2681 Erweiterung bereitstellt]

2682 \RequirePackage{tikz-uml}
2683 \RequirePackage{tikz-uml-activity}
2684 % Not compatible with wasysym
2685 %\RequirePackage{mathabx}
2686 \RequirePackage{wasysym}
2687 \usetikzlibrary{positioning}

2688 \tikzumlset{
2689   fill class=white!0,
2690   font=\footnotesize,
2691   fill object=white!0,
2692   fill note=white!0,
2693   fill state=white!0,
2694   % Use case
2695   fill usecase=white!0,
2696   fill system=white!0,
2697 }
```



```
\liUmlLeserichtung \umluniassoc[arg1=,mult2=1,arg2=- druckerei,name=benutzt]{TicketHandel}{TicketDruckerei}
\liUmlLeserichtung[pos=below left,dir=down,distance=0cm]{benutzt}
```

```

2698 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2699   \def\@liDirLeft{}
2700   \def\@liDirRight{ \RIGHTarrow}
2701   \pgfkeys{/lese/dir/.is choice}
2702   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2703   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2704   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2705   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2706
2707   \def\@liPos{above}
2708   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2709

```

```

2710 \def\@liDistance{0cm}
2711 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2712
2713 \pgfkeys{/lese/.cd,#1}
2714
2715 \node[\@liPos = \@liDistance of #2-middle] {
2716   \@liDirLeft{\footnotesize#2}\@liDirRight
2717 };
2718 }
2719

```

## 2.44 vollstaendige-induktion.sty

2720 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2721 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01  
 2722 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die  
 2723 Überschriften für die einzelnen Schritte]

### Let-Abkürzungen

\let\m=\liInduktionMarkierung  
 \let\e=\liInduktionErklaerung

```
\begin{align*}
C_{n+1}
&= \frac{
  \{(4 \cdot (\{m\{n+1\} - 1) + 2) \cdot \text{cn}(\{m\{n+1\} - 1)\}
  \{m\{n+1\} + 1\}
  & \text{\textit{Java nach Mathe}}\} \\
& \\
&= \frac{
  \{(4\{m\{n\} + 2) \cdot \text{cn}(\{m\{n\}\}
  \{m\{n+2\}\}
  & \text{\textit{addiert, subtrahiert}}\} \\
& \\
&= \frac{
  \{(4n+2) \cdot \{m\{(2n)\}\}
  \{(n+2) \cdot \{m\{(n+1)\} \cdot n!\}
  & \text{\textit{für cn(n) Formel eingesetzt}}\} \\
& \\
&= \frac{
  \{(4n+2) \cdot (2n)! \cdot \{m\{\cdot (n+1)\}\}
  \{(n+2) \cdot (n+1)! \cdot n! \cdot \{m\{\cdot (n+1)\}\}
  & \text{\textit{\$(n+1)\$ multipliziert}}\} \\
& \\
&= \frac{
  \{(4n+2) \cdot \{m\{(n+1) \cdot (2n)\}\}
  \{(n+2) \cdot (n+1)! \cdot \{m\{(n+1) \cdot n!\}
  & \text{\textit{umsortiert}}\} \\
& \\
&= \frac{
  \{m\{(2(n+1))!\}
  \{m\{(n+2)! \cdot (n+1)!\}
  & \text{\textit{Hilfsgleichungen verwendet}}\} \\
& \\
&= \frac{
  \{(2(\{m\{n+1\}\})!\}
  \{((\{m\{n+1\} + 1)! \cdot (\{m\{n+1\}!\}
  & \text{\textit{\$(n+1)\$ verdeutlicht}}\} \\
& \\
& \\
\end{align*}
```

Lade häufig benötigte Pakete

2724 \RequirePackage{lehramt-informatik-typographie}  
 2725 \RequirePackage{lehramt-informatik-mathe}  
 2726 \RequirePackage{lehramt-informatik-syntax}  
 2727 \ExplSyntaxOn

\liInduktionMarkierung Hilfsmakro um Teile von mathematischen Formeln markieren zu können.

**Let-Abkürzung:** \let\m=\liInduktionMarkierung

2728 \def\liInduktionMarkierung#1{\textcolor{violet}{#1}}

\liInduktionErklaerung Gedacht für die rechte Spalte in der align-Umgebung. Das text-Makro ist dann nicht mehr nötig.

**Let-Abkürzung:** \let\e=\liInduktionErklaerung

```
2729 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2730 \def\liInduktionAnfang{
2731   \liPseudoUeberschrift{Induktionsanfang}
2732
2733   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2734   \liParagraphMitLinien{
2735     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
2736   }
2737 }
```

\liInduktionVoraussetzung

```
2738 \def\liInduktionVoraussetzung{
2739   \liPseudoUeberschrift{Induktionsvoraussetzung}
2740
2741   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2742   \liParagraphMitLinien{
2743     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
2744   }
2745 }
```

\liInduktionSchritt

```
2746 \def\liInduktionSchritt{
2747   \liPseudoUeberschrift{Induktionsschritt}
2748
2749   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2750   \liParagraphMitLinien{
2751     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
2752     auch~ $A(n=k+1)$ ~wahr~sein~muss.
2753   }
2754 }
```

```
2755 \ExplSyntaxOff
```

```
2756
```



## 2.45 wasserfall.sty

```
2757 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2758 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2759 \RequirePackage{tikz}
2760 \tikzset{wasserfall/.style={
2761   >=stealth,
2762   node distance = 2mm and -8mm,
2763   start chain = A going below right,
2764   every node/.style = {
2765     draw,
2766     text width=24mm,
2767     minimum height=12mm,
2768     align=center,
2769     inner sep=1mm,
2770     fill=white,
2771     drop shadow={fill=black},
2772     on chain=A
2773   },
2774 }}
2775 \usetikzlibrary{chains,positioning,shadows}
2776
```

## 2.46 wpkalkuel.sty

2777 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2778 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

### Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2779 \RequirePackage{amsmath}

2780 \ExplSyntaxOn

\liWpKalkuel **Let-Abkürzung:** \let\wp=\liWpKalkuel

2781 \def\liWpKalkuelOhneMathe#1#2{

2782 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2783 }

2784 \def\liWpKalkuel#1#2{

2785 \ifmmode

2786 \liWpKalkuelOhneMathe{#1}{#2}

2787 \else

2788 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2789 \fi

2790 }

\MatheEnv

2791 \def\MatheEnv#1{

2792 \medskip

2793

2794 \hspace{1em}#1

2795

2796 \medskip

2797 }

\Mathe

2798 \def\Mathe#1{

2799 \MatheEnv{#1\$}

2800 }

\liWpEquivalent **Let-Abkürzung:** \let\equivalent=\liWpEquivalent

2801 \def\liWpEquivalent#1{

2802 \MatheEnv{\$\equiv\$\hspace{1em}#1\$}

2803 }

\liWpErklaerung **Let-Abkürzung:** \let\erklaerung=\liWpErklaerung

2804 \newlength{\@Skip@Erklaerung@Reset}

2805 \def\liWpErklaerung#1{

2806 \setlength{\@Skip@Erklaerung@Reset}{\leftskip}

2807 \setlength{\leftskip}{0.5cm}

2808

2809 \par

2810 \noindent

2811 {

2812 \scriptsize

2813 #1

2814 }

2815 \par

2816

2817 \setlength{\leftskip}{\@Skip@Erklaerung@Reset}

2818 }

`\liWpErklaerungVerzweigung`

```

2819 \def\liWpErklaerungVerzweigung{
2820   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2821   \equiv
2822   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2823   \lor
2824   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2825 }

2826 \ExplSyntaxOff

2827

```

### 3 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

	<b>Symbols</b>	<code>\AfterEndEnvironment</code> 2478	2190, 2192, 2193, 2562
<code>\#</code> .....	109	<code>\allsectionsfont</code> ...	1231
<code>\,</code> ....	343, 398, 1182, 1184, 1685, 1686, 1687, 2015, 2485, 2670	<code>\Alph</code> .....	1245
<code>\@Skip@Erklaerung@Reset</code> ... ..	2804, 2806, 2817	<code>\alph</code> .....	1245, 1246
<code>\@afterheading</code> .....	1652	<code>\alpha</code> 2585, 2587, 2588, 2591, 2593, 2594, 2595, 2596, 2597,	
<code>\@afterindentfalse</code> .	1652	2601, 2607, 2608, 2613, 2614, 2615,	
<code>\@liDirLeft</code> 2699, 2704, 2716		2618, 2626, 2627, 2628	
<code>\@liDirRight</code> 2700, 2702, 2703, 2704, 2705, 2716		<code>\arabic</code> 1245, 2204, 2209, 2214, 2220, 2226, 2232	
<code>\@liDistance</code> .....	2710, 2711, 2715	<code>\arraystretch</code> .....	1911
<code>\@liPos</code> ..	2707, 2708, 2715		
<code>\</code> .....	608, 631, 632, 635, 636, 639, 640, 732, 733, 734, 841, 870, 872, 898, 907, 952, 994, 995, 996, 1001, 1002, 1003, 1023, 1559, 1916, 1974, 1977	<b>B</b>	
<code>\{</code> .....	207, 1119, 1129, 1141, 1142, 1147, 1181, 1400, 1961, 2360, 2670, 2820	<code>\BeforeBeginEnvironment</code> .....	2477
<code>\}</code> .....	207, 1119, 1129, 1141, 1142, 1149, 1185, 1401, 1961, 2360, 2674, 2820	<code>\begin</code> ..	629, 680, 695, 730, 754, 801, 833, 848, 868, 878, 894, 914, 946, 963, 992, 1017, 1038, 1053, 1162, 1251, 1326, 1354, 1363, 1370, 1478, 1551, 1603, 1608, 1616, 1641, 1646, 1657, 1667, 1671, 1754, 1758, 1774, 1797, 1820, 1835, 1914, 1915, 1965, 2004, 2124, 2144, 2285, 2307, 2321, 2477, 2488, 2674
<code>\_</code> 27, 35, 44, 46, 293, 318, 342, 343, 357, 358, 364, 367, 370, 380, 392		<code>\begingroup</code> 1552, 1963, 2405	
<code>\_</code> .....	1973, 1981	<code>\beschriftung</code> .....	1583, 1587, 1591, 1595, 1599, 1601
		<code>\beta</code> .....	2585, 2588, 2592, 2593, 2594, 2597, 2607, 2608, 2609, 2614, 2615
		<code>\bf</code> .....	2182, 2183, 2184
		<code>\bfseries</code> ..	487, 1235, 1237, 2182, 2188,
<b>A</b>			
<code>\addbibresource</code> .....	1529, 1530, 1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538		
<code>\advance</code> .....	2409		
		<b>C</b>	
		<code>\c</code> .....	1328, 1329
		<code>\cdot</code> ....	1729, 1782, 1793
		<code>\centerline</code> .....	1359, 2283, 2305, 2320
		<code>\chapter</code> .....	1235, 1236
		<code>\char</code> .....	1559
		<code>\clearpage</code> .....	1553
		<code>\cline</code> .....	608
		<code>\clist</code> .....	226, 270, 271, 290, 296, 2417
		<code>\columnbreak</code> .....	2452
		<code>\cs</code> ....	293, 318, 342, 343, 380, 392, 1635
		<code>\csname</code> .....	1313, 1316
		<code>\cup</code> .....	1142, 1990, 2594, 2608, 2615
		<b>D</b>	
		<code>\DeclareMathSymbol</code> ..	1859, 1860
		<code>\DecoINERT</code> .....	2207, 2280, 2281, 2303
		<code>\DecoINERTwithPivot</code> .	2223, 2302
		<code>\DecoLEFT</code> .....	2202, 2301
		<code>\DecoLEFTwithPivot</code> ..	2217, 2279
		<code>\DecoRIGHT</code> 2212, 2304, 2314	
		<code>\DecoRIGHTwithPivot</code> .	2229, 2282
		<code>\definecolor</code> .....	1233
		<code>\delta</code> 65, 107, 165, 207, 1133	



2141, 2151, 2585, 2587, 2593, 2614, 2743	\inhaltsverzeichnis <a href="#">1550</a>	\input . 4, 7, 10, 13, 16, 406	\inputminted <a href="#">2495, 2505, 2515, 2527, 2530, 2534</a>	\int ..... <a href="#">2417</a>	\item ..... <a href="#">475, 476, 697, 701, 706, 711, 755, 764, 769, 777, 849, 854, 858, 879, 915, 920, 927, 935, 964, 969, 973, 978, 1054, 1059, 1064, 1364, 1365, 1635, 1639, 1759, 1762, 1766, 1775, 1781, 1786, 1798, 1802, 1806, 1810, 1814, 1821, 1825, 1829, 2125, 2128, 2131, 2145, 2148, 2151</a>	\itshape ..... <a href="#">531, 2634</a>	<b>J</b>	\j <a href="#">1328, 1329, 1331, 1332, 1333, 1338, 1339, 1340</a>	<b>K</b>	\k ..... <a href="#">1338</a>	\keys ..... <a href="#">31, 70, 82, 112, 122, 170, 180, 298, 550, 554, 568, 573, 1206, 1213</a>	<b>L</b>	\l ..... <a href="#">63, 64, 65, 66, 67, 68, 71, 72, 73, 74, 75, 77, 79, 84, 85, 86, 87, 88, 89, 104, 105, 106, 107, 108, 109, 110, 113, 114, 115, 116, 117, 118, 119, 125, 126, 127, 128, 129, 130, 131, 162, 163, 164, 165, 166, 167, 168, 171, 172, 173, 174, 175, 176, 177, 183, 184, 185, 186, 187, 188, 189, 548, 551, 556, 557, 560, 565, 566, 569, 570, 575, 1201, 1202, 1203, 1204, 1207, 1208, 1209, 1210, 1216, 1217, 1218, 1219, 1494, 1495, 1496, 1638, 1639, 1640, 1647</a>	\labelenumi ..... <a href="#">1246</a>	\labelenumii ..... <a href="#">1247</a>	\labelitemi ..... <a href="#">1240</a>	\labelitemii ..... <a href="#">1241</a>	\labelitemiii ..... <a href="#">1242</a>	\labelitemiv ..... <a href="#">1243</a>	\land ..... <a href="#">2822, 2824</a>	\LARGE ..... <a href="#">1235</a>	\large ..... <a href="#">1359, 2394</a>	\leaders ..... <a href="#">2663</a>	\left ..... <a href="#">1694</a>	\LEFTarrow ..... <a href="#">2704</a>	\leftarrow ..... <a href="#">594</a>	\leftouterjoin ..... <a href="#">2333</a>	\leftskip <a href="#">2806, 2807, 2817</a>	\LehramtInformatikAutorEmail ..... <a href="#">1510</a>	\LehramtInformatikAutorName ..... <a href="#">1509</a>	\LehramtInformatikGitBranch ..... <a href="#">385, 2471</a>	\LehramtInformatikGithubCodeRepository ..... <a href="#">2470</a>	\LehramtInformatikGithubDomain ..... <a href="#">2467</a>	\LehramtInformatikGithubRawDomain ..... <a href="#">383, 2468</a>	\LehramtInformatikGithubTexRepository ..... <a href="#">384, 2469</a>	\LehramtInformatikRepository ..... <a href="#">4, 7, 10, 13, 16, 1529, 1530, 1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538, 2466</a>	\LehramtInformatikTitel ..... <a href="#">1505</a>	\leq .... <a href="#">1793, 2128, 2148</a>	\let ..... <a href="#">1131, 1132, 1553, 2279, 2280, 2281, 2282, 2295, 2297, 2298, 2299, 2301, 2302, 2303, 2304, 2314, 2406, 2444, 2445, 2655</a>	\li@chomsky@erklaerung@texte ..... <a href="#">497, 533</a>	\li@EntwurfsCode ..... <a href="#">625, 671, 672, 673, 719, 720, 721, 722, 788, 789, 790, 791, 792, 793, 820, 821, 822, 823, 824, 825, 886</a>	\li@EntwurfsCodeAllgemein ..... <a href="#">624</a>	\li@fussnote@text <a href="#">1867, 1873, 1877, 1881, 1885</a>	\li@GithubLink ..... <a href="#">2487, 2500, 2510, 2521</a>	\li@mget . <a href="#">1315, 1319, 1339</a>	\li@minc ..... <a href="#">1318, 1340</a>	\li@mset ..... <a href="#">1312, 1320, 1329, 1332</a>	\li@numdiscs ..... <a href="#">1324, 1333, 1339</a>	\li@Rmodell@Schrift . <a href="#">2344, 2353, 2363</a>	\li@sequence .. <a href="#">1325, 1338</a>	\li@synthese@erklaerung@texte ..... <a href="#">2576, 2636</a>	\liAbleitung ..... <a href="#">1156</a>	liAdditum (environment) ..... <a href="#">1606</a>	liAHuelle (environment) ..... <a href="#">1962</a>	\liAlphabet ..... <a href="#">1141</a>	liAntwort (environment) ..... <a href="#">1577</a>	\liAnweisung ..... <a href="#">1485</a>	\liAssemblerCode ... <a href="#">2525</a>	\liAssemblerDatei .. <a href="#">2526</a>	\liAttribut ..... <a href="#">2363</a>	\liAttributHuelle ... <a href="#">1953, 2588, 2593</a>	\liAttributHuelleOhneMathe ..... <a href="#">1953, 1956,</a>	\liAttributMenge .... <a href="#">1958, 1972, 1980, 1985</a>	\liAufgabe ..... <a href="#">3</a>	\liAufgabenMetadaten . <a href="#">25</a>	\liAufgabenTitel .... <a href="#">23</a>	\liAusdruck ..... <a href="#">1178</a>	\liAutomat ..... <a href="#">61</a>	\liAutomatenKante .... <a href="#">93</a>	\liBandAlphabet .... <a href="#">1142</a>	\liBedingung ..... <a href="#">1486</a>	\liBedingungDrei .... <a href="#">1749, 1789, 1829</a>	\liBedingungEins .... <a href="#">1743, 1778, 1821</a>	\liBedingungFalsch . <a href="#">1488</a>	\liBedingungWahr ... <a href="#">1487</a>	\liBedingungZwei .... <a href="#">1746, 1784, 1825</a>	\liBeschriftung .... <a href="#">1566</a>	\liChomskyErklaerung ..... <a href="#">497, 538</a>	\liChomskyUeberErklaerung ..... <a href="#">536</a>	\liChomskyUeberschrift ..... <a href="#">485, 537</a>	\liCpmEreignis ..... <a href="#">546</a>	\liCpmFruehesterI ... <a href="#">603</a>	\liCpmSpaetesterI ... <a href="#">602</a>	\liCpmVon ..... <a href="#">586</a>	\liCpmVonOhneMathe .. <a href="#">586, 589, 591</a>	\liCpmVonZu ..... <a href="#">578</a>	\liCpmVonZuOhneMathe ..... <a href="#">578, 581, 583</a>	\liCpmVorgang ..... <a href="#">563</a>	\liCpmZu ..... <a href="#">594</a>	\liCpmZuOhneMathe ... <a href="#">594, 597, 599</a>	liDiagramm (environ- ment) ..... <a href="#">1665</a>	liEinbettung (environ- ment) ..... <a href="#">1576</a>	\liEntwurfs ..... <a href="#">1033</a>
---	--	--------------------------------	---	---------------------------------	--	--	----------	---	----------	-------------------------------	---	----------	---	--	---	--	---	--	---	--	-----------------------------------	---	-------------------------------------	----------------------------------	---------------------------------------	--------------------------------------	---	--	---	--	---	---	---	---	---	---	--	--	---	---	--	---	--	---	---	---	---	---	--	--	--	---	--	--	--	--	---	---	---	--	--	--	--	------------------------------------	---	--	--	-------------------------------------	---	---	---	--	--	---	---	--	---	---	---	---	--	---	---	-------------------------------------	---	---------------------------------------	--	---	------------------------------------	---	--	--	--

\liEntwurfsAbstrakteFabrik	\liEntwurfsModellPraesentation	\liKellerKante	102
..... 675	..... 1016, 1034	\liKellerKante	140
\liEntwurfsAbstrakteFabrikCode	\liEntwurfsZustand	\liKellerUebergang	135, 141
..... 670, 677	\liEntwurfsZustandAkteure	\liKontrollCode	1489
\liEntwurfsAbstrakteFabrikUml	..... 1052, 1072	\liKontrollflussgraph	1477
..... 628, 676	\liEntwurfsZustandUml	(environment)	1491
\liEntwurfsAdapter	..... 1037, 1071	\liKontrollKnotenPfad	1491
\liEntwurfsAdapterAkteure	\liEpsilon	\liKontrollTextzeileKnoten	1490, 1495
..... 694, 726	\liErAttribute	\liKurzeTabellenLinie	608
\liEntwurfsAdapterCode	... 1085, 1099, 1101	\liLadeAllePakete	228
..... 718, 727	\liErDatenbankName	\liLadePakete	54, 57, 224, 229, 484, 545, 1118, 1349, 1865, 1945, 2087, 2558
\liEntwurfsAdapterUml	\liErEntity	\liLatexCode	2486
..... 679, 725	\liErledigt	\liLeereZelle	1903
\liEntwurfsBeobachter	..... 2655	\liLernkartei (environ-	1655
\liEntwurfsBeobachterAkteure	\liErMpAttribute	ment)	1970
..... 753, 797	... 1098	\liLinksReduktion	1970
\liEntwurfsBeobachterCode	\liErMpEntity	\liLinksReduktionInline	1979, 1984
..... 787, 798	..... 1086	\liMasterExkurs	1834
\liEntwurfsBeobachterUml	\liErMpRelationship	\liMasterFaelle	1773, 1841
..... 729, 796	\liErRelationship	\liMasterFallRechnung	1819
\liEntwurfsDekorierer	... 1084, 1093, 1095	\liMasterVariablen	1753, 1836
\liEntwurfsDekoriererAkteure	\liExamensAufgabe	\liMasterVariablenDeklaration	1796
..... 829	..... 6	\liMasterWolframLink	1844
\liEntwurfsDekoriererCode	\liExamensAufgabeA	\liMenge	71, 72, 74, 113, 114, 115, 119, 171, 172, 173, 177, 1119, 1168, 1207, 1208
..... 819, 830	\liExamensAufgabeTA	\liMinimierungErklaerung	1925
\liEntwurfsDekoriererUml	\liExamensAufgabeTTA	\liMinispracheDatei	2529
..... 800, 828	.. 9	\linespread	2351
\liEntwurfsEinfacheFabrik	\liExkurs (environment)	\liNichtsZuTun	2656
..... 863	..... 1614	\liO	1716, 1744
\liEntwurfsEinfacheFabrikAkteure	\liFalsch	\liOmega	1706, 1750
..... 847, 865	\liFlaci	\liOmegaOhneMathe	1706, 1711, 1713
\liEntwurfsEinfacheFabrikUml	..... 1188	\liOOhneMathe	1716, 1721, 1723
..... 832, 864	\liFremd	\liParagraphMitLinien	533, 1927, 2636, 2657, 2734, 2742, 2750
\liEntwurfsEinzelstueck	\liFunktionaleAbhaengigkeit	\liPetriErreichKnotenDrei	2081
..... 888	... 1987, 1990, 1997	\liPetriErreichTransition	2078
\liEntwurfsEinzelstueckAkteure	\liFunktionaleAbhaengigkeiten	\liPetriSetzeSchluessel	2023
..... 877, 890	..... 2000	\liPetriTransitionsName	2070, 2082
\liEntwurfsEinzelstueckCode	\liFussnote	\liPetriTransitionsNameOhneMathe	2070, 2073, 2075
..... 885, 891	... 1866, 1868		
\liEntwurfsEinzelstueckUml	\liFussnoteDreiText		
..... 867, 889	..... 1880, 1898		
\liEntwurfsErbauer	\liFussnoteEinsText		
\liEntwurfsErbauerAkteure	..... 1872, 1892		
..... 913, 943	\liFussnoteLink		
\liEntwurfsErbauerUml	..... 1682		
..... 893, 942	\liFussnoten		
\liEntwurfsFabrikmethode	\liFussnoteUrl		
..... 987, 1012	... 1031, 1678		
\liEntwurfsFabrikmethodeAkteure	\liFussnoteVierText		
..... 962, 989	..... 1884, 1901		
\liEntwurfsFabrikmethodeUml	\liFussnoteZweiText		
..... 945, 988	..... 1876, 1895		
\liEntwurfsKompositum	\liGeschweifteKlammern		
..... 1012	... 1160, 2001, 2667		
\liEntwurfsKompositumAkteure	\liGrammatik		
..... 1014	..... 1199		
\liEntwurfsKompositumUml	\liGraphenFormat (envi-		
..... 991, 1013	ronment)		
\liEntwurfsModellPraesentation	\liHanoi		
..... 1033	..... 1312		
\liEntwurfsModellPraesentation	\liHaskellCode		
..... 1035	..... 2532		
	\liHaskellDatei		
	... 2533		
	\liInduktionAnfang		
	... 2730		
	\liInduktionErklaerung		
	..... 2729		
	\liInduktionMarkierung		
	..... 2728		
	\liInduktionSchritt		
	... 2746		
	\liInduktionVoraussetzung		
	..... 2738		
	\liJavaCode		
	..... 2485		
	\liJavaDatei		
	... 626, 2494		
	\liJavaExamen		
	..... 2514		
	\liJavaTestAkteure		
	... 2504		
	\liKasten (environment)		
	..... 1250		

<code>\liPetriTransPfeile</code>	<a href="#">2082</a>	<code>\liTheta</code>	<a href="#">1696</a> , <a href="#">1747</a> , <a href="#">1776</a> , <a href="#">1782</a> , <a href="#">1787</a>	<code>\loop</code>	<a href="#">2164</a>
<code>\liPolynomiellReduzierbar</code>	<a href="#">1369</a>	<code>\liThetaOhneMathe</code>	<a href="#">1696</a> , <a href="#">1701</a> , <a href="#">1703</a>	<code>\lor</code>	<a href="#">2823</a>
<code>\liPotenzmenge</code>	<a href="#">1128</a> , <a href="#">1132</a> , <a href="#">2093</a>	<code>\liTOhneMathe</code>	<a href="#">1726</a> , <a href="#">1735</a> , <a href="#">1737</a>	<code>\ltimes</code>	<a href="#">1852</a>
<code>\liPotenzmengeOhneMathe</code>	<a href="#">1129</a> , <a href="#">1130</a> , <a href="#">1131</a>	<code>\liTuringKante</code>	<a href="#">203</a>	<b>M</b>	
<code>\liPrimaer</code>	<a href="#">2342</a>	<code>\liTuringLeerzeichen</code>	<a href="#">159</a> , <a href="#">167</a>		
<code>\liProblemBeschreibung</code>	<a href="#">1353</a>	<code>\liTuringMaschine</code>	<a href="#">160</a>	<code>\makeatletter</code>	<a href="#">1651</a> , <a href="#">2403</a>
<code>\liProblemClique</code>	<a href="#">1376</a>	<code>\liTuringUeberfuehrung</code>	<a href="#">206</a>	<code>\makeatother</code>	<a href="#">1653</a> , <a href="#">2411</a>
<code>\liProblemName</code>	<a href="#">1352</a> , <a href="#">1359</a> , <a href="#">1371</a> , <a href="#">1373</a> , <a href="#">1386</a> , <a href="#">1397</a> , <a href="#">1398</a> , <a href="#">1406</a> , <a href="#">1407</a>	<code>\liTuringUebergaenge</code>	<a href="#">198</a> , <a href="#">204</a>	<code>\marginpar</code>	<a href="#">1088</a> , <a href="#">1094</a> , <a href="#">1100</a> , <a href="#">1558</a>
<code>\liProblemSat</code>	<a href="#">1405</a>	<code>\liTuringUebergangZelle</code>	<a href="#">193</a>	<code>\mathbb</code>	<a href="#">1407</a> , <a href="#">2151</a> , <a href="#">2743</a>
<code>\liProblemSubsetSum</code>	<a href="#">1396</a> , <a href="#">1405</a>	<code>\liUeberfuehrungsFunktion</code>	<a href="#">1133</a>	<code>\mathbin</code>	<a href="#">2333</a> , <a href="#">2334</a> , <a href="#">2335</a>
<code>\liProblemVertexCover</code>	<a href="#">1376</a> , <a href="#">1384</a>	<code>\liUeberfuehrungsFunktionOhneMathe</code>	<a href="#">1133</a> , <a href="#">1136</a> , <a href="#">1138</a>	<code>\mathcal</code>	<a href="#">1717</a> , <a href="#">2614</a> , <a href="#">2619</a> , <a href="#">2621</a> , <a href="#">2622</a> , <a href="#">2623</a>
<code>\liProduktionen</code>	<a href="#">1167</a> , <a href="#">1209</a>	<code>liUebergangsTabelle</code>	<a href="#">1911</a>	<code>\Mathe</code>	<a href="#">2798</a>
<code>liProduktionsRegeln</code>	(environment) <a href="#">1157</a>	<code>\liUeberschriftDreiecksTabelle</code>	<a href="#">1921</a>	<code>\MatheEnv</code>	<a href="#">2791</a> , <a href="#">2799</a> , <a href="#">2802</a>
<code>liProjektSprache</code>	(environment) <a href="#">1575</a>	<code>\liUmlLeserichtung</code>	<a href="#">2698</a>	<code>\mathord</code>	<a href="#">1859</a> , <a href="#">1860</a>
<code>\liPseudoUeberschrift</code>	<a href="#">1560</a> , <a href="#">1610</a> , <a href="#">1611</a> , <a href="#">1913</a> , <a href="#">1923</a> , <a href="#">2731</a> , <a href="#">2739</a> , <a href="#">2747</a>	<code>\liVertauschen</code>	<a href="#">2371</a>	<code>\mdfsetup</code>	<a href="#">1249</a> , <a href="#">1584</a> , <a href="#">1588</a> , <a href="#">1592</a> , <a href="#">1596</a>
<code>\liPumpingKontextfrei</code>	<a href="#">2139</a>	<code>\liWortInSprache</code>	<a href="#">609</a>	<code>\medskip</code>	<a href="#">1361</a> , <a href="#">1569</a> , <a href="#">1571</a> , <a href="#">1645</a> , <a href="#">1673</a> , <a href="#">2349</a> , <a href="#">2355</a> , <a href="#">2665</a> , <a href="#">2792</a> , <a href="#">2796</a>
<code>\liPumpingRegulaer</code>	<a href="#">2118</a>	<code>\liWortNichtInSprache</code>	<a href="#">614</a>	<code>\emph</code>	<a href="#">1558</a>
<code>liQuellen</code>	(environment) <a href="#">1635</a>	<code>\liWpEquivalent</code>	<a href="#">2801</a>	<code>\mintinline</code>	<a href="#">2485</a> , <a href="#">2486</a> , <a href="#">2525</a> , <a href="#">2532</a> , <a href="#">2537</a>
<code>\liRechtsReduktionInline</code>	<a href="#">1984</a>	<code>\liWpErklaerung</code>	<a href="#">2804</a>	<code>\mkern</code>	<a href="#">2333</a> , <a href="#">2334</a> , <a href="#">2335</a>
<code>\liRekursionsGleichung</code>	<a href="#">1740</a> , <a href="#">1800</a>	<code>\liWpErklaerungVerzweigung</code>	<a href="#">2819</a>	<code>\mlq</code>	<a href="#">1857</a> , <a href="#">1859</a>
<code>\liRelation</code>	<a href="#">2011</a>	<code>\liWpKalkuel</code>	<a href="#">2781</a>	<code>\mrq</code>	<a href="#">1857</a> , <a href="#">1860</a>
<code>liRelationenSchemaFormat</code>	(environment) <a href="#">2364</a>	<code>\liWpKalkuelOhneMathe</code>	<a href="#">2781</a> , <a href="#">2786</a> , <a href="#">2788</a> , <a href="#">2820</a> , <a href="#">2822</a> , <a href="#">2824</a>	<code>\msg</code>	<a href="#">39</a> , <a href="#">403</a>
<code>\liRelationMenge</code>	<a href="#">2358</a>	<code>\liZustandsBuchstabe</code>	<a href="#">1143</a> , <a href="#">1152</a> , <a href="#">1154</a> , <a href="#">1172</a> , <a href="#">1174</a>	<code>\myList</code>	<a href="#">2424</a> , <a href="#">2425</a> , <a href="#">2426</a> , <a href="#">2429</a>
<code>\liRichtig</code>	<a href="#">475</a>	<code>\liZustandsBuchstabeGross</code>	<a href="#">1144</a> , <a href="#">1153</a> , <a href="#">1155</a>	<code>\myNodes</code>	<a href="#">2413</a> , <a href="#">2428</a> , <a href="#">2434</a> , <a href="#">2438</a> , <a href="#">2440</a>
<code>liRmodell</code>	(environment) <a href="#">2344</a>	<code>\liZustandsmenge</code>	<a href="#">1131</a>	<b>N</b>	
<code>\liRundeKlammer</code>	<a href="#">1693</a> , <a href="#">1697</a> , <a href="#">1707</a> , <a href="#">1717</a> , <a href="#">1731</a>	<code>\liZustandsmengeNr</code>	<a href="#">1145</a> , <a href="#">2102</a>		
<code>\liSortierMarkierung</code>	<a href="#">2383</a>	<code>\liZustandsmengeNrGross</code>	<a href="#">1153</a>	<code>\NeedsTeXFormat</code>	<a href="#">1</a> , <a href="#">19</a> , <a href="#">52</a> , <a href="#">220</a> , <a href="#">410</a> , <a href="#">471</a> , <a href="#">478</a> , <a href="#">542</a> , <a href="#">605</a> , <a href="#">620</a> , <a href="#">1075</a> , <a href="#">1111</a> , <a href="#">1224</a> , <a href="#">1264</a> , <a href="#">1273</a> , <a href="#">1278</a> , <a href="#">1307</a> , <a href="#">1345</a> , <a href="#">1417</a> , <a href="#">1500</a> , <a href="#">1515</a> , <a href="#">1521</a> , <a href="#">1543</a> , <a href="#">1689</a> , <a href="#">1849</a> , <a href="#">1862</a> , <a href="#">1941</a> , <a href="#">2018</a> , <a href="#">2084</a> , <a href="#">2109</a> , <a href="#">2114</a> , <a href="#">2158</a> , <a href="#">2324</a> , <a href="#">2337</a> , <a href="#">2366</a> , <a href="#">2447</a> , <a href="#">2454</a> , <a href="#">2459</a> , <a href="#">2539</a> , <a href="#">2554</a> , <a href="#">2645</a> , <a href="#">2649</a> , <a href="#">2678</a> , <a href="#">2720</a> , <a href="#">2757</a> , <a href="#">2777</a>
<code>\liSortierPfeil</code>	<a href="#">2377</a>	<code>\liZustandsMengenSammlung</code>	<a href="#">2089</a>	<code>\neg</code>	<a href="#">2824</a>
<code>\liSortierPfeilUnten</code>	<a href="#">2380</a>	<code>\liZustandsMengenSammlungNr</code>	<a href="#">2098</a>	<code>\negthinspace</code>	<a href="#">1694</a>
<code>\liSpaltenUmbruch</code>	<a href="#">2452</a>	<code>\liZustandsmengeOhneMathe</code>	<a href="#">1131</a>	<code>\newcounter</code>	<a href="#">2178</a> , <a href="#">2179</a>
<code>\liSqlCode</code>	<a href="#">2537</a>	<code>\liZustandsname</code>	<a href="#">1154</a>	<code>\NewDocumentCommand</code>	<a href="#">62</a> , <a href="#">103</a> , <a href="#">140</a> , <a href="#">161</a> , <a href="#">203</a> , <a href="#">224</a> , <a href="#">547</a> , <a href="#">564</a> , <a href="#">609</a> , <a href="#">614</a> , <a href="#">1178</a> , <a href="#">1200</a> , <a href="#">1369</a> , <a href="#">1492</a> , <a href="#">1519</a> , <a href="#">1678</a> , <a href="#">1682</a> , <a href="#">2000</a> , <a href="#">2011</a> , <a href="#">2078</a> , <a href="#">2494</a> , <a href="#">2504</a> , <a href="#">2514</a> , <a href="#">2526</a> , <a href="#">2529</a> , <a href="#">2533</a> , <a href="#">2698</a>
<code>\listen@punkt</code>	<a href="#">1635</a> , <a href="#">1647</a>	<code>\liZustandsnameGross</code>	<a href="#">1155</a> , <a href="#">2090</a> , <a href="#">2099</a>	<code>\NewDocumentEnvironment</code>	<a href="#">1157</a> , <a href="#">1250</a> , <a href="#">1305</a> ,
<code>\liStrich</code>	<a href="#">1351</a>	<code>\liZustandsnameTiefgestellt</code>	<a href="#">1170</a>		
<code>\liSyntheseErklaerung</code>	<a href="#">2576</a> , <a href="#">2641</a>	<code>\liZustandsPaar</code>	<a href="#">1905</a>		
<code>\liSyntheseUeberErklaerung</code>	<a href="#">2639</a>	<code>\liZustandsPaarVariablenName</code>	<a href="#">1904</a> , <a href="#">1907</a> , <a href="#">1908</a>		
<code>\liSyntheseUeberschrift</code>	<a href="#">2560</a> , <a href="#">2640</a>	<code>\llap</code>	<a href="#">2331</a>		
<code>\liT</code>	<a href="#">1726</a> , <a href="#">1741</a> , <a href="#">1755</a> , <a href="#">1816</a>	<code>\log</code>	<a href="#">1744</a> , <a href="#">1747</a> , <a href="#">1750</a> , <a href="#">1776</a> , <a href="#">1782</a>		
<code>\liTeilen</code>	<a href="#">1950</a>				
<code>\literatur</code>	<a href="#">1517</a> , <a href="#">1541</a>				



1477, 1575, 1576, 1579, 1606, 1614, 1636, 1655, 1665, 1912, 1962, 2346, 2364	\newlength ..... 2804	\ProvidesPackage .... 2, 20, 53, 221, 411, 472, 479, 543, 606, 621, 1076, 1112, 1225, 1265, 1274, 1279, 1308, 1346, 1418, 1501, 1516, 1522, 1544, 1690, 1850, 1863, 1942, 2019, 2085, 2110, 2115, 2159, 2325, 2338, 2367, 2448, 2455, 2460, 2540, 2555, 2646, 2650, 2679, 2721, 2758, 2778	474, 544, 623, 624, 1078, 1080, 1081, 1117, 1226, 1227, 1230, 1232, 1234, 1239, 1248, 1255, 1266, 1267, 1276, 1280, 1281, 1282, 1310, 1311, 1350, 1419, 1518, 1523, 1524, 1540, 1547, 1548, 1549, 1577, 1692, 1853, 1854, 2020, 2022, 2112, 2172, 2174, 2326, 2327, 2328, 2341, 2369, 2400, 2451, 2457, 2462, 2473, 2474, 2542, 2647, 2654, 2682, 2683, 2685, 2686, 2724, 2725, 2726, 2759, 2779
\node ..... 560, 1485, 1490, 2204, 2209, 2214, 2220, 2226, 2232, 2383, 2428, 2715	\noexpand ..... 2292, 2293, 2294, 2313, 2428	<b>Q</b>	\right ..... 1694
\noindent ..... 355, 611, 616, 1562, 1564, 1568, 1572, 1600, 1628, 1630, 1643, 1659, 1661, 1669, 1838, 1891, 1894, 1897, 1900, 2359, 2658, 2669, 2810	\nolinkurl ..... 2491	\QS@list ..... 2285, 2296, 2300, 2307, 2313, 2318, 2321	\RIGHTarrow ... 2700, 2705
\normalsize ..... 1237	\notin ..... 617	\QS@select@equal .... ..... 2260, 2264	\Rightarrow .... 612, 617
\null ..... 2663	<b>O</b>	\QS@select@greater .. ..... 2261, 2265	\rightarrow ..... 207, 501, 506, 514, 518, 520, 521, 523, 578, 586, 2082, 2585, 2592, 2594, 2597, 2602, 2607, 2608, 2613
\o@join ..... 2330, 2333, 2334, 2335	\Omega ..... 1707	\QS@select@smaller .. ... 2256, 2259, 2263	\rightouterjoin .... 2334
\omega 2120, 2121, 2141, 2142	\or ..... 2241	\QS@sort@a ..... 2238, 2271, 2292, 2293	\Roman ..... 1245
<b>P</b>	\pagestyle ..... 1260	\QS@sort@b .... 2238, 2239	\roman ..... 1245, 1247
\par ..... 354, 375, 537, 1189, 1197, 1567, 1629, 1652, 1660, 2283, 2305, 2320, 2361, 2640, 2664, 2668, 2809, 2815	\paragraph ..... 1237	\QS@sort@c .... 2242, 2249	\romannumeral ..... 2250
\parindent ..... 2352	\path .... 94, 141, 204, 575	\QS@sort@d .... 2250, 2258	\rtimes ..... 1852
\pgfkeys .. 2034, 2701, 2702, 2703, 2704, 2705, 2708, 2711, 2713	\pgfmath@count ..... ... 2405, 2407, 2409	\QS@sort@empty . 2240, 2245	\rule 2283, 2305, 2320, 2331
\pgfmath@smuggleone 2410	\pgfmathdeclarefunction ..... 2404	\QS@sort@single 2241, 2246	<b>S</b>
\pgfmathhint ..... 2405	\pgfmathparse ..... ..... 1319, 2422, 2427, 2430, 2444, 2445	\QSinitialize ..... ... 2163, 2275, 2310	\sb ..... 67, 77, 79, 108, 166, 513, 514, 518, 521, 522, 523, 1172, 1174, 1744, 1747, 1750, 1776, 1782, 1931, 2070, 2079, 2607, 2608, 2609, 2614, 2618, 2619, 2622, 2623, 2626, 2627, 2628
\pgfmathresult ..... ..... 1320, 2405, 2406, 2408, 2410, 2423, 2431, 2444, 2445	\pgfutil@empty ..... 2406	\QSIr . 2246, 2252, 2260, 2280, 2294, 2299, 2302	\scriptscriptstyle .. ..... 578, 586, 594
\pgfutil@loop ..... 2407	\pgfutil@repeat .... 2410	\QSIrr 2281, 2294, 2295, 2303	\scriptsize .... 1191, 1430, 1437, 1443, 1505, 1506, 1509, 1510, 2729, 2782, 2812
\preceq ..... 1372	\prime ..... 1351	\QSLr ..... 2252, 2259, 2270, 2271, 2279, 2292, 2297, 2301	\section ..... 46
\printbibliography . 1541		\QSPivotStep ..... 2165, 2275, 2279, 2290	\seq . 1494, 1495, 1496, 1638, 1639, 1640, 1647
		\QSRr ..... 2261, 2282, 2293, 2298, 2304, 2313, 2314, 2315	\setbox ..... 2330
		\QSsortStep ..... 2167, 2275, 2291, 2292	\setcounter ..... 1238, 2284, 2306, 2320
		\quad ..... 1869	\setganttlinklabel .. 1268, 1269, 1270, 1271
	<b>R</b>	<b>R</b>	\setlength ..... 2352, 2806, 2807, 2817
	\raisebox ..... 1490	\relax ..... 1553, 2252, 2295, 2297, 2298, 2299, 2407, 2409	
	\renewcommand ... 1240, 1241, 1242, 1243, 1246, 1247, 1261, 1262, 1511, 1512, 1911	\repeat ..... 2168	
	\RequirePackage ..... ... 55, 158, 222, 226, 407, 413, 414,		



<code>\setmainfont</code> . . . . .	1228	<code>\titlespacing</code> . . . . .	1236	731, 736, 741, 744,
<code>\setmainlanguage</code> . . . .	408	<code>\tl</code> . . . . .	29, 37, 63, 64, 65,	802, 803, 804, 809,
<code>\setminted</code> . . . . .	2479, 2480		66, 67, 68, 71, 72,	810, 839, 869, 896,
<code>\setminus</code> . . . . .	1986		73, 74, 75, 77, 79,	897, 900, 951, 954,
<code>\setsansfont</code> . . . . .	1229		104, 105, 106, 107,	993, 999, 1000,
<code>\setul</code> . . . . .	2343		108, 109, 110, 113,	1018, 1019, 1020,
<code>\sffamily</code> . . . . .	488,		114, 115, 116, 117,	1039, 1040, 1041, 1042
	1235, 1237, 1339, 2563		118, 119, 162, 163,	<code>\umld</code> . . . . .
<code>\shoveleft</code> . . . . .	1971		164, 165, 166, 167,	<code>\umldHagg</code> . . . . .
<code>\shoveright</code> . . . . .	1975		168, 171, 172, 173,	
<code>\Sigma</code> . . . . .	64, 105,		174, 175, 176, 177,	749, 815, 1009
	163, 1141, 1142, 1202		291, 295, 319, 323,	<code>\umlinherit</code> . . . . .
<code>\sigma</code> . . . . .	511, 513, 514		324, 325, 328, 333,	688, 739, 904, 949, 957
<code>\SLASH</code> . . . . .	1559		334, 335, 346, 347,	<code>\uml</code> . . . . .
<code>\small</code> . . . . .	1668		348, 349, 360, 366,	<code>\umlreal</code> . . . . .
<code>\sort</code> . . . . .	2417		369, 372, 381, 395,	686, 747
<code>\sortList</code> . . . . .	2416, 2425		548, 551, 556, 557,	<code>\umlsimpleclass</code> . .
<code>\square</code> . . . . .	476		565, 566, 569, 570,	645,
<code>\stepcounter</code> 2204, 2209,			1180, 1201, 1202,	646, 647, 651, 653,
2214, 2217, 2219,			1203, 1204, 1207,	654, 655, 681, 834,
2223, 2225, 2229, 2231			1208, 1209, 1210, 1727	835, 836, 895, 947, 948
<code>\str</code> . . . . .	489, 498, 1581,	<code>\tmp</code> . . . . .	1988	<code>\umlstatic</code> . . . . .
	2094, 2103, 2564, 2577	<code>\TmpPlaceEight</code> . . . . .	2042	841, 870
<code>\string</code> . . . . .	1973, 1981	<code>\TmpPlaceFive</code> . . . . .	2039	<code>\umluniagg</code> . . . . .
<code>\StrSubstitute</code> . . . . .	2424, 2426	<code>\TmpPlaceFour</code> . . . . .	2038	902
<code>\strut</code> . . . . .	1798, 1802,	<code>\TmpPlaceNine</code> . . . . .	2043	<code>\umluniassoc</code> . . . .
	1806, 1810, 1814, 2452	<code>\TmpPlaceOne</code> . . . . .	2035	665,
<code>\subseq</code> 2588, 2621, 2628		<code>\TmpPlaceSeven</code> . . . . .	2041	687, 903, 1027, 1028
		<code>\TmpPlaceSix</code> . . . . .	2040	<code>\umlVHuniassoc</code> . .
		<code>\TmpPlaceTen</code> . . . . .	2044	666, 667
		<code>\TmpPlaceThree</code> . . . . .	2037	<code>\umlVHvdep</code> . . . . .
		<code>\TmpPlaceTwo</code> . . . . .	2036	659,
		<code>\TmpScale</code> . . . . .	2055	660, 662, 663, 843, 844
		<code>\TmpTransitionEight</code> .	2031, 2052	<code>\umlVHVinherit</code> . . . . .
		<code>\TmpTransitionFive</code> . .	2028, 2049	642, 643, 648, 649,
		<code>\TmpTransitionFour</code> . .	2027, 2048	656, 657, 812, 813,
		<code>\TmpTransitionNine</code> . .	2032, 2053	837, 838, 1007, 1008
		<code>\TmpTransitionOne</code> . . .	2024, 2045	<code>\umlVHVreal</code> . . . . .
		<code>\TmpTransitionSeven</code> .	2030, 2051	806, 807, 1044, 1045
		<code>\TmpTransitionSix</code> . . .	2029, 2050	<code>\UParrow</code> . . . . .
		<code>\TmpTransitionTen</code> . . .	2033, 2054	2702
		<code>\TmpTransitionThree</code> .	2026, 2047	<code>\url</code> . . . . .
		<code>\TmpTransitionTwo</code> . . .	2025, 2046	1679
		<code>\TmpX</code> . . . . .	2056	<code>\usemintedstyle</code> . . . .
		<code>\TmpY</code> . . . . .	2057	2476
		<code>\today</code> . . . . .	1506	<code>\usetikzlibrary</code> . .
		<code>\ttfamily</code> . . . . .	2344	56,
				415, 1079, 1283,
				1420, 2021, 2329,
				2370, 2401, 2687, 2775
				V
				<code>\value</code> . . . . .
				2166
				<code>\varepsilon</code> . . . . .
				490,
				501, 502, 1127,
				1744, 1750, 1779, 1790
				<code>\vfill</code> . . . . .
				2452
				<code>\vrule</code> . . . . .
				2659, 2663
				<code>\vspace</code> . . . . .
				1615, 1633, 2671, 2673
	</			

\xintLength	.....	2238		<b>Z</b>		\zustandsnamens@liste
			\ZB	.....	<u>1686</u>	... 1145, 1152, 1153
\xintnthelt	.....	2250	\zB	.....	<u>1685</u>	