

# lehramt-informatik

Hermine Bschlangaul \*

August 31, 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}
ZitatSchluessel = sosy:pu:4,
ZitatBeschreibung = {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 \_gib_github_url:
47
48 \section{\_gib_aufgaben_titel:}
49
50 \bigskip
51 }

52 \ExplSyntaxOff

53
```

## 2.3 automaten.sty

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

### 2.3.1 Endlicher Automat

```
56 \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}
```



```
57 \RequirePackage{tikz}
58 \usetikzlibrary{arrows,automata,positioning}
59 \liLadePakete{mathe}
60 \directlua{
61   automaten = require('lehramt-informatik-automaten')
62 }
```

`\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}`:  $A_{NEA} = (Z, \Sigma, \delta, E, z_0)$

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

```

77     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
78     dea .value_forbidden:n = true,
79     dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
80     nea .value_forbidden:n = true,
81     nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
82   }
83
84   \keys_set:nn { automat } { #2 }
85
86   $ #1 \l_typ_tl = (
87     \l_zustaende_tl,
88     \l_alphabet_tl,
89     \l_delta_tl,
90     \l_ende_tl,
91     \l_start_tl
92   )$
93 }
94 \ExplSyntaxOff

```

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

```

95 \def\liAutomatenKante#1#2#3#4{
96   \path (#1) edge[#4] node{#3} (#2);
97 }

98 \tikzset{
99   li automat/.style={
100     ->,
101     node distance=2cm
102   },
103 }

```

### 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]
{\langle zustaeende=Z,alphabet=Sigma,kelleralphabet=Gamma,delta=delta,start=z0,kellerboden=#,ende=E \rangle}

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

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

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

```

120     kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
121     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
122   }
123
124   \keys_set:nn { kellerautomat } { #2 }
125
126   $#1 = (
127     \l_zustaende_tl,
128     \l_alphabet_tl,
129     \l_kelleralphabet_tl,
130     \l_delta_tl,
131     \l_start_tl,
132     \l_kellerboden_tl,
133     \l_ende_tl
134   )$
135 }
136 \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}`

```

137 \ExplSyntaxOn
138 \def\liKellerUebergang#1{
139   \directlua{automaten.drucke_keller_uebergaenge('#1')}
140 }
141 \ExplSyntaxOff

```

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

```

142 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
143   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
144 }

145 \tikzset{
146   li keller knoten/.style={
147     text width=2cm,
148     align=center,
149     font=\footnotesize,
150   },
151   li kellerautomat/.style={
152     li automat,
153     every edge/.append style={
154       every node/.style={
155         li keller knoten
156       }
157     }
158   }
159 }

```

### 2.3.3 Turingmaschine

```
160 \RequirePackage{amssymb}
```

**\liTuringLeerzeichen**

□

```
161 \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},
}

```

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

```

162 \ExplSyntaxOn
163 \NewDocumentCommand {\liTuringMaschine} { O{TM} m } {
164   \tl_set:Nn \l_zustaende_tl {Z}
165   \tl_set:Nn \l_alphabet_tl {\Sigma}
166   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
167   \tl_set:Nn \l_delta_tl {\delta}
168   \tl_set:Nn \l_start_tl {z\sb{0}}
169   \tl_set:Nn \l_leerzeichen_tl {\liTuringLeerzeichen}
170   \tl_set:Nn \l_ende_tl {E}
171
172   \keys_define:nn { kellerautomat } {
173     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
174     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
175     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\liMenge{##1}}},
176     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
177     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
178     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
179     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
180   }
181
182   \keys_set:nn { kellerautomat } { #2 }
183
184   $\text{#1} = (
185     \l_zustaende_tl,
186     \l_alphabet_tl,
187     \l_bandalphabet_tl,
188     \l_delta_tl,
189     \l_start_tl,
190     \l_leerzeichen_tl,
191     \l_ende_tl
192   )$
193 }
194 \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)`

```

195 \ExplSyntaxOn
196 \def\liTuringUebergangZelle#1{
197   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
198 }
199 \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)`

```

200 \ExplSyntaxOn
201 \def\liTuringUebergaenge#1{
202   \directlua{automaten.drucke_turing_uebergaenge('#1')}
203 }
204 \ExplSyntaxOff

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

205 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
206   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
207 }

\liTuringUeberfuehrung

208 \def\liTuringUeberfuehrung{
209    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
210 }

211 \tikzset{
212   li turingmaschine/.style={
213     li automat,
214     every edge/.append style={
215       every node/.style={
216         li keller knoten
217       }
218     }
219   }
220 }
221

```

## 2.4 basis.sty

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

224 \RequirePackage{xparse}

225 \ExplSyntaxOn

\liLadePakete

226 \NewDocumentCommand{\liLadePakete}{ m }
227 {
228   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
229 }

\liLadeAllePakete

230 \def\liLadeAllePakete{
231   \liLadePakete{
232     aufgaben-einbinden,
233     automaten,
234     baum,
235     checkbox,
236     chomsky-normalform,
237     cpm,
238     cyk-algorithmus,
239     entwurfsmuster,
240     er,
241     formale-sprachen,
242     gantt,
243     grafik,
244     graph,
245     hanoi,
246     kontrollflussgraph,
247     makros,
248     master-theorem,
249     mathe,
250     minimierung,
251     normalformen,
252     petri,
253     potenzmengen-konstruktion,
254     pseudo,
255     quicksort,
256     relationale-algebra,
257     rmodell,
258     sortieren,
259     spalten,
260     struktogramm,
261     syntax,
262     syntaxbaum,
263     synthese-algorithmus,
264     tabelle,
265     typographie,
266     uml,
267     vollstaendige-induktion,
268     wasserfall,
269     wpkalkuel,
270   }
271 }
```

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.

```
272 \clist_new:N \g_auf_schluessel_clist
273 \clist_set:Nn \g_auf_schluessel_clist {
```



```

274  titel,
275  thematik,
276  stichwoerter,
277  zitat_schluessel,
278  zitat_beschreibung,
279  %
280  bearbeitungs_stand,
281  korrektheit,
282  %
283  relativer_pfad,
284  %
285  examen_nummer,
286  examen_jahr,
287  examen_monat,
288  examen_thema_nr,
289  examen_telaufgabe_nr,
290  examen_aufgabe_nr,
291 }

```

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

```

292 \clist_map_inline:Nn \g_auf_schluessel_clist {
293   \tl_new:c {g_auf_#1_tl}
294 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

295 \cs_new:Npn \setze_variablen_zurueck: {
296   \clist_map_inline:Nn \g_auf_schluessel_clist {
297     \tl_clear:c {g_auf_##1_tl}
298   }
299 }

```

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

```

300 \keys_define:nn { aufgabenmetadaten }
301 {
302   Titel .tl_gset:N = \g_auf_titel_tl,
303   Thematik .tl_gset:N = \g_auf_thematik_tl,
304   Stichwoerter .tl_gset:N = \g_auf_stichwoerter_tl,
305   ZitatSchluessel .tl_gset:N = \g_auf_zitat_beschreibung_tl,
306   ZitatBeschreibung .tl_gset:N = \g_auf_zitat_schluessel_tl,
307   %
308   BearbeitungsStand .tl_gset:N = \g_auf_bearbeitungs_stand_tl,
309   Korrektheit .tl_gset:N = \g_auf_korrekttheit_tl,
310   %
311   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
312   %
313   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
314   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
315   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
316   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
317   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_telaufgabe_nr_tl,
318   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
319 }
320 \cs_gset:Npn \setze_relativen_pfad: {
321   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
322   {
323     \bool_if:nTF
324     {
325       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
326       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
327       ! \tl_if_empty_p:N \g_auf_examen_monat_tl
328     }
329     {

```

```

330 \tl_gset:Nn \g_auf_relativer_pfad_tl {
331   Staatsexamen /
332   \g_auf_examen_nummer_tl /
333   \g_auf_examen_jahr_tl /
334   \g_auf_examen_monat_tl /
335   \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
336   \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_te
337   \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr
338 }
339 }
340 {}
341 }
342 {}
343 }

344 \cs_gset:Npn \_gib_examen_titel: {
345   \cs_set:Nn \_trenner: { \, / \, }
346   \bool_if:nTF
347   {
348     ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
349     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
350     ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
351     ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
352   }
353   {
354     {
355       \footnotesize
356       \par
357       \noindent
358       Staatsexamen ~
359       \g_auf_examen_nummer_tl \_trenner:
360       \g_auf_examen_jahr_tl \_trenner:
361
362       \tl_case:Nn \g_auf_examen_monat_tl
363       {
364         { 03 } { Frühjahr }
365         { 09 } { Herbst }
366       } \_trenner:
367
368       \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
369         Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
370       }
371       \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
372         Teilaufgabe ~ Nr. ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
373       }
374       \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
375         Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
376       }
377       \par
378       \bigskip
379     }
380   }
381 }

382 \cs_new:Npn \_gib_github_url: {
383   \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
384     Github :~\href{
385       \LehramtInformatikGithubRawDomain /
386       \LehramtInformatikGithubTexRepo /
387       \LehramtInformatikGitBranch /
388       \g_auf_relativer_pfad_tl
389     }{
390       \g_auf_relativer_pfad_tl
391     }
392   }

```

```

393 }
394 \cs_new:Npn \_gib_aufgaben_titel: {
395   \g_auf_titel_tl
396
397   \tl_if_empty:NTF \g_auf_thematik_tl
398   {}
399   {
400     \, ~ [
401       \g_auf_thematik_tl
402     ]
403   }
404 }
405 \msg_new:nnn { aufgabenmetadaten } { kein-titel }
406   { Der~Schlüssel~ist~zwingend~notwendig }
407 \ExplSyntaxOff
408 \def\LehramtInformatikRepository{/pfad/zum/repository}
409 \input /etc/lehramt-informatik.config.tex
410 biblatex not working with lualatex and babel
411 % \RequirePackage{polyglossia}
412 % \setmainlanguage{german}
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500

```

## 2.5 baum.sty

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

415 \RequirePackage{tikz}

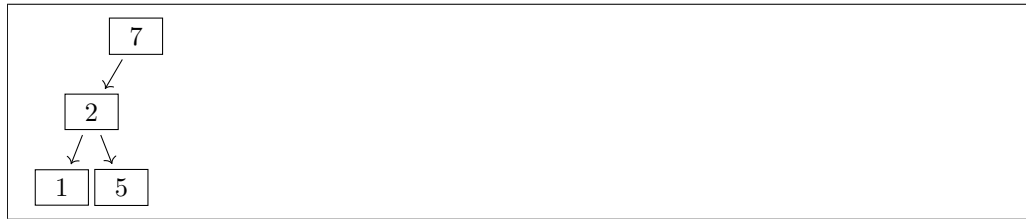
    für li binaer baum
416 \RequirePackage{tikz-qtree}

    Für b baum
417 \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}
```



```
418 \tikzset{
419   li binaer baum/.style={
420     shorten <=2pt,
421     shorten >=2pt,
422     ->,
423     every tree node/.style={
424       minimum width=2em,
425       draw,
426       rectangle
427     },
428     blank/.style={
429       draw=none
430     },
431     edge from parent/.style={
432       draw,
433       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
434     },
435     level distance=1cm,
436     every label/.style={
437       gray,
438       font=\footnotesize,
439       label position=0,
440       label distance=0cm,
441     }
442   },
443 }
```

### 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}

```



```

444 \tikzset{
445   li bbaum knoten/.style={
446     rectangle split parts=10,
447     rectangle split,
448     rectangle split horizontal,
449     rectangle split ignore empty parts,
450     draw,
451     fill=white
452   },
453   li bbaum/.style={
454     every node/.style={
455       li bbaum knoten
456     },
457     level 1/.style={
458       level distance=12mm,
459       sibling distance=25mm,
460     },
461     every child/.style={
462       shorten <= 2pt,
463       shorten >= 6pt,
464       ->,
465     },
466     level 2/.style={
467       level distance=9mm,
468       sibling distance=15mm,
469     },
470   }
471 }
472

```

## 2.6 checkbox.sty

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

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

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

479
```



## 2.7 chomsky-normalform.sty

```

480 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
481 \ProvidesPackage{lehramt-informatik-chomsky-normalform}[2021/03/26
482 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]
483
484 \ExplSyntaxOn
485
486 \liLadePakete{typographie}

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

487 \def\liChomskyUeberschrift#1{
488   {
489     \bfseries
490     \sffamily
491     \str_case:nn {#1} {
492       {1} {Elimination~der~ $\varepsilon$ -Regeln}
493       {2} {Elimination~von~Kettenregeln}
494       {3} {Separation~von~Terminalzeichen}
495       {4} {Elimination~von~mehrelementigen~Nonterminalketten}
496     }
497   }
498 }

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

499 \def\li@chomsky@erklaerung@texte#1{
500   \str_case:nn {#1} {
501     %
502     {1} {
503       Alle~Regeln~der~Form~ $A \rightarrow \varepsilon$ ~werden~eliminiert.~
504       Die~Ersetzung~von~ $A$ ~wird~durch~ $\varepsilon$ ~in~allen~anderen~
505       Regeln~vorweggenommen.
506     }
507     {2} {
508       Jede~Produktion~der~Form~ $A \rightarrow B$ ~mit~ $A, B \in S$ ~wird~
509       als~Kettenregel~bezeichnet.~Diese~tragen~nicht~zur~Produktion~
510       von~Terminalzeichen~bei~und~lassen~sich~ebenfalls~eliminieren.
511     }
512     {3} {
513       Jedes~Terminalzeichen~ $\sigma$ ,~das~in~Kombination~mit~anderen~
514       Symbolen~auftaucht,~wird~durch~ein~neues~Nonterminal~
515        $S_{\sigma}$ ~ersetzt~und~die~Menge~der~Produktionen~durch~die~
516       Regel~ $S_{\sigma} \rightarrow \sigma$ ~ergänzt.
517     }
518     {4} {
519       Alle~Produktionen~der~Form~
520        $A \rightarrow B_1 B_2 \dots B_n$ ~
521       werden~in~die~Produktionen~
522        $A \rightarrow$ 
523        $A_{n-1} B_{n-1}, A_{n-1} \rightarrow$ 
524        $A_{n-2} B_{n-2}, \dots,$ 
525        $A_2 \rightarrow B_1 B_2$ ~zerteilt.~
526       Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
527       vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
528     }
529   }
530 }

531 \def\liChomskyErklaerung#1{
532   {
533     \itshape
534     \footnotesize
535     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
536   }

```

537 }

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

538 \def\liChomskyUeberErklaerung#1{

539 \liChomskyUeberschrift{#1}\par

540 \liChomskyErklaerung{#1}

541 }

542 \ExplSyntaxOff

543

## 2.8 cpm.sty

```

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

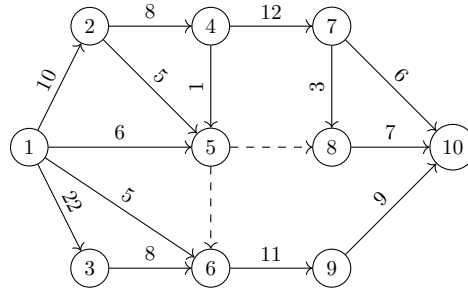
```

### 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}{ }
\liCpmVorgang[schein]{5}{8}{ }
\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}

```

```

548 \ExplSyntaxOn
549 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
550   \tl_set:Nn \l_name_tl {}
551
552   \keys_define:nn { cpmEreignis } {
553     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
554   }
555
556   \keys_set:nn { cpmEreignis } { #1 }
557
558   \tl_if_empty:NT \l_name_tl {
559     \tl_set:Nn \l_name_tl {#2}
560   }

```

```

561
562 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
563 }
564 \ExplSyntaxOff

\liCpmVorgang liCpmVORGANG\((.*)>(.*))\{(.*)\}
565 \ExplSyntaxOn
566 \NewDocumentCommand { \liCpmVorgang } { 0{ } m m m } {
567   \tl_set:Nn \l_schein_tl {}
568   \tl_set:Nn \l_kritisch_tl {}
569
570   \keys_define:nn { cpmVorgang } {
571     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
572     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
573   }
574
575   \keys_set:nn { cpmVorgang } { #1 }
576
577   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
578 }
579 \ExplSyntaxOff

```

### 2.8.1 Hilfsmakros für Zeitpunkt-Tabelle

```

\begin{tabular}{|l||l|l|l|l|l|l|l|l|}
\hline
$i$ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & \\
\hline
\FZ & 0 & 5 & 18 & 7 & 19 & 26 & 22 & 30 & \\
\hline
\SZ & 0 & 5 & 18 & 9 & 19 & 26 & 24 & 30 & \\
\hline
GP & 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & \\
\hline
\end{tabular}

```

```

\liCpmVonZu Let-Abkürzung: \let\vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $l_{(2 \rightarrow 3)}$ 
580 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
581 \def\liCpmVonZu#1(#2-#3){%
582   \ifmmode%
583     \liCpmVonZuOhneMathe{#1}(#2-#3)%
584   \else%
585     $\liCpmVonZuOhneMathe{#1}(#2-#3)$%
586   \fi%
587 }

```

```

\liCpmVon Let-Abkürzung: \let\v=\liCpmVon
\liCpmVon{1}(2):  $l_{(\rightarrow 2)}$ 
588 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
589 \def\liCpmVon#1(#2){%
590   \ifmmode%
591     \liCpmVonOhneMathe{#1}(#2)%
592   \else%
593     $\liCpmVonOhneMathe{#1}(#2)$%
594   \fi%
595 }

```

```

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

```

```

602 \fi%
603 }

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

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

606

```

## 2.9 cyk-algorithmus.sty

```
607 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
608 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
609 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`

```
610 \def\liKurzeTabellenLinie#1{\ccline{1-#1}}
```

`\liWortInSprache`  $\Rightarrow abc \in L(Y)$

```
611 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
612   \bigskip
613   \noindent
614   $\Rrightarrow #1 \in #2$
615 }
```

`\liWortNichtInSprache`  $\Rightarrow abc \notin L(G)$

```
616 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
617   \bigskip
618   \noindent
619   $\Rrightarrow #1 \notin #2$
620 }
```

```
621
```

## 2.10 entwurfsmuster.sty

```
622 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
623 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06
624 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

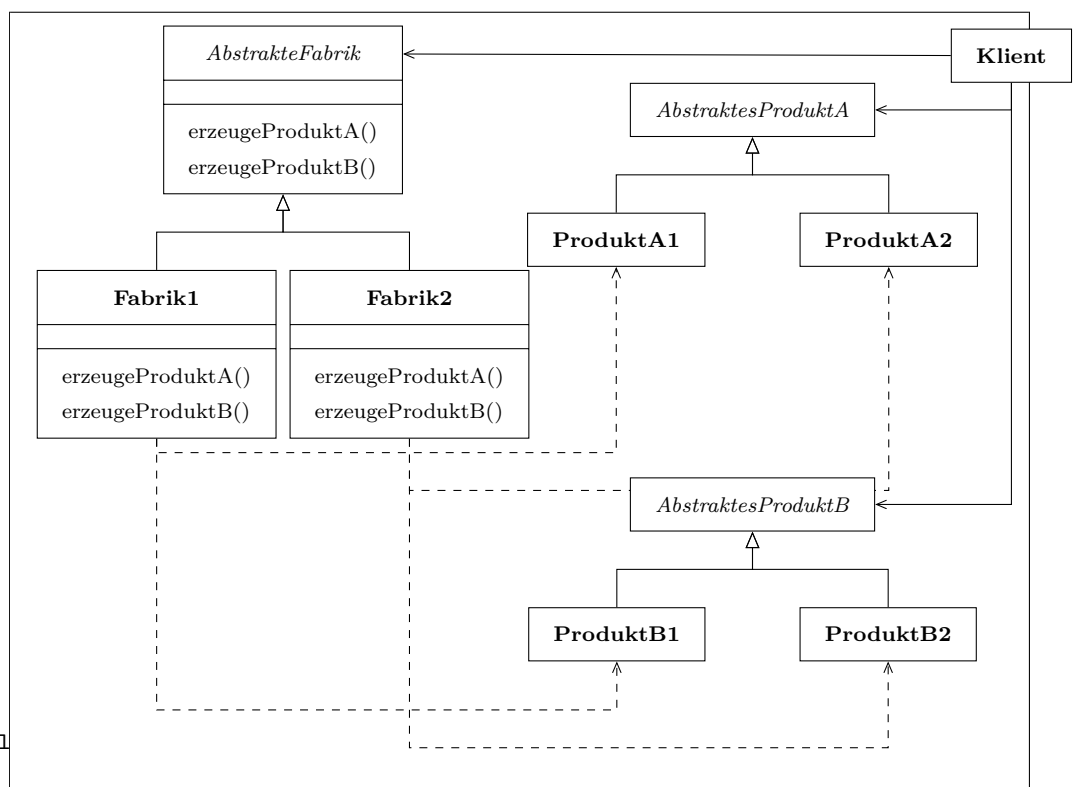
1. Uml: Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. Akteure: Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. Code: Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

- EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
626 \RequirePackage{lehramt-informatik-syntax}
627 \def\li@EntwurfsCode#1#2{
628   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
629 }
```

### 2.10.3 Abstrakte Fabrik (Abstract Factory)



```

630 \def\liEntwurfsAbstrakteFabrikUml{
631   \begin{tikzpicture}
632     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
633       erzeugeProduktA()\}
634       erzeugeProduktB()\}
635   }

```

```

636 \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
637     erzeugeProduktA()\
638     erzeugeProduktB()\
639 }
640 \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
641     erzeugeProduktA()\
642     erzeugeProduktB()\
643 }
644 \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
645 \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
646
647 \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
648 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
649 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
650 \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
651 \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
652
653 \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
654
655 \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
656 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
657 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
658 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
659 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
660
661 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
662 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
663
664 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
665 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
666
667 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
668 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
669 \umlVHuniassoc{Klient}{AbstraktesProduktB}
670 \end{tikzpicture}
671 }

```

iEntwurfsAbstrakteFabrikCode

```

672 \def\liEntwurfsAbstrakteFabrikCode{
673     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
674     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
675     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
676 }

```

\liEntwurfsAbstrakteFabrik

```

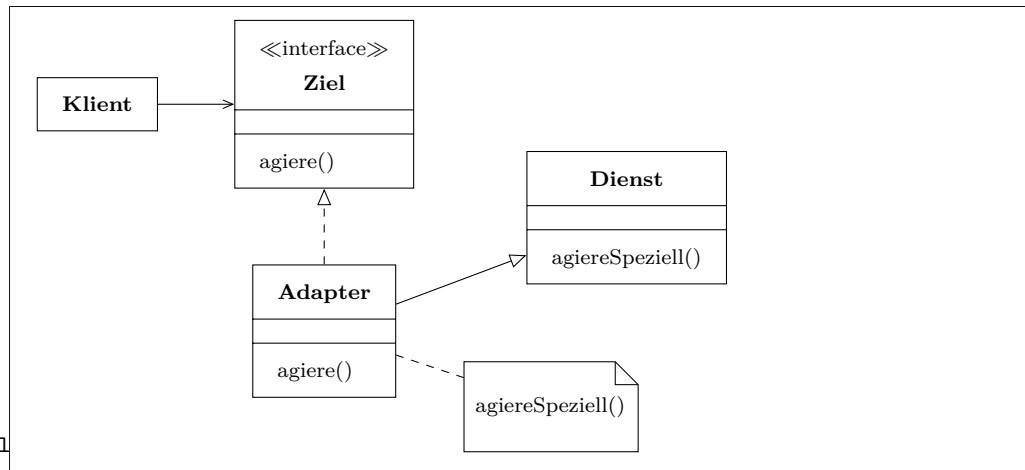
677 \def\liEntwurfsAbstrakteFabrik{
678     \liEntwurfsAbstrakteFabrikUml
679     \liEntwurfsAbstrakteFabrikCode
680 }

```

## 2.10.4 Adapter



\liEntwurfsAdapterUml



```

681 \def\liEntwurfsAdapterUml{
682   \begin{tikzpicture}
683     \umlsimpleclass[x=1,y=3]{Klient}{}{}
684     \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
685     \umlclass[x=4,y=0]{Adapter}{}{agiere()}
686     \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
687
688     \umlreal{Adapter}{Ziel}
689     \umluniassoc{Klient}{Ziel}
690     \umlinherit{Adapter}{Dienst}
691
692     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
693   \end{tikzpicture}
694   \footcite[so ähnlich wie GoF]{wiki:adapter}
695 }

```

\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.

```

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

```

```

717
718 \end{description}
719 }

```

\liEntwurfsAdapterCode

```

720 \def\liEntwurfsAdapterCode{
721 \li@EntwurfsCode{adapter}{Dienst}
722 \li@EntwurfsCode{adapter}{Ziel}
723 \li@EntwurfsCode{adapter}{Adapter}
724 \li@EntwurfsCode{adapter}{Klient}
725 }

```

\liEntwurfsAdapter

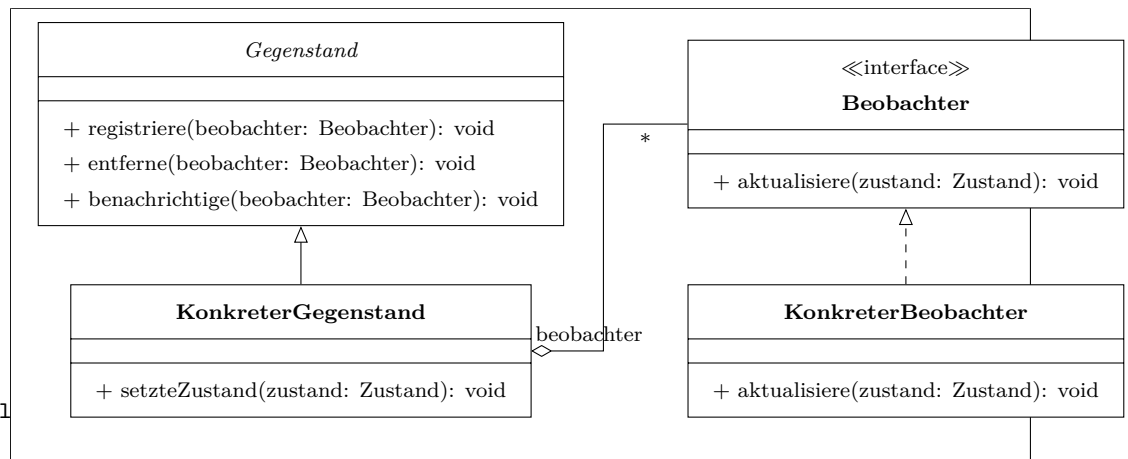
```

726 \def\liEntwurfsAdapter{
727 \liEntwurfsAdapterUml
728 \liEntwurfsAdapterAkteure
729 \liEntwurfsAdapterCode
730 }

```

### 2.10.5 Beobachter (Observer)

\liEntwurfsBeobachterUml



```

731 \def\liEntwurfsBeobachterUml{
732 \begin{tikzpicture}
733 \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
734 + registriere(beobachter: Beobachter): void\\
735 + entferne(beobachter: Beobachter): void\\
736 + benachrichtige(beobachter: Beobachter): void\\
737 }
738 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
739 + setzteZustand(zustand: Zustand): void
740 }
741 \umlinherit{KonkreterGegenstand}{Gegenstand}
742
743 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
744 + aktualisiere(zustand: Zustand): void
745 }
746 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
747 + aktualisiere(zustand: Zustand): void
748 }
749 \umlreal{KonkreterBeobachter}{Beobachter}
750
751 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
752 {KonkreterGegenstand}{Beobachter}
753 \end{tikzpicture}
754 }

```

\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.

```
755 \def\liEntwurfsBeobachterAkteure{
756   \begin{description}
757     \item[Gegenstand / Subjekt (Subject / Observable)]
758
759     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
760     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
761     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
762     und Abmeldung von Beobachtern und eine Schnittstelle zur
763     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
764     251]{gof}
765
766     \item[Beobachter (Observer)]
767
768     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
769     genannt) definieren eine Aktualisierungsschnittstelle.
770
771     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
772
773     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
774     den relevanten Zustand und benachrichtigt alle Beobachter bei
775     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
776     verfügt über eine Schnittstelle zur Erfragung des aktuellen
777     Zustands.
778
779     \item[Konkrete Beobachter (ConcreteObserver)]
780
781     Konkrete Beobachter verwalten die Referenz auf ein konkretes
782     Subjekt, dessen Zustand sie beobachten und speichern und dessen
783     Zustand konsistent ist. Sie implementieren eine
784     Aktualisierungsschnittstelle unter Verwendung der
785     Abfrageschnittstelle des konkreten Subjekts.
786     \footcite{wiki:beobachter}
787   \end{description}
788 }
```

\liEntwurfsBeobachterCode

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

```

795 \li@EntwurfsCode{beobachter}{Klient}
796 }

```

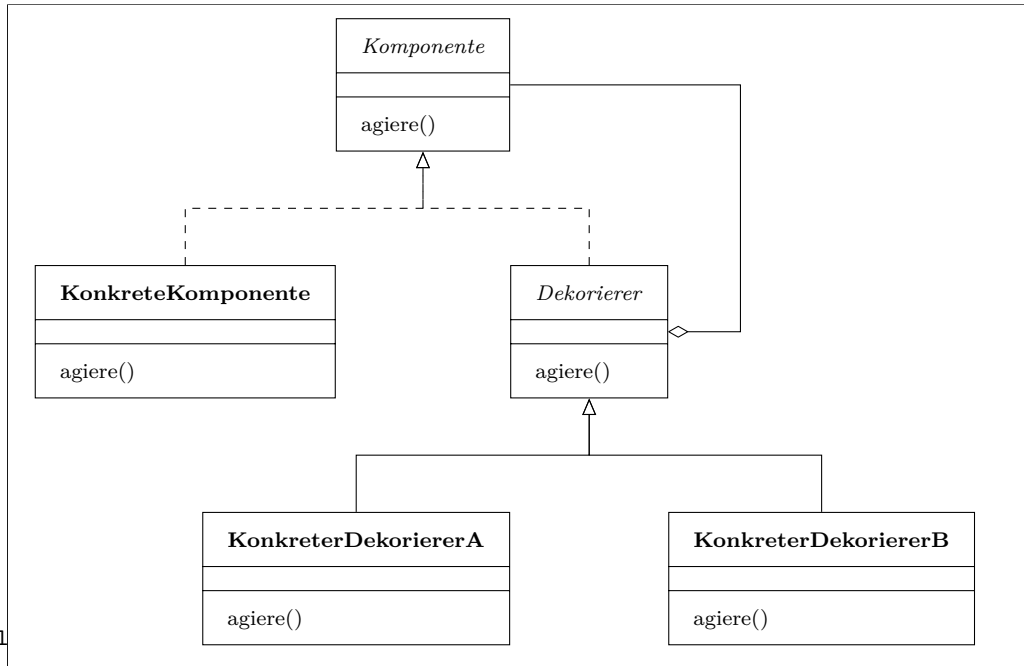
\liEntwurfsBeobachter

```

797 \def\liEntwurfsBeobachter{
798 \liEntwurfsBeobachterUml
799 \liEntwurfsBeobachterAkteure
800 \liEntwurfsBeobachterCode
801 }

```

## 2.10.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

802 \def\liEntwurfsDekoriererUml{
803 \begin{tikzpicture}
804 \umlclass[type=abstract]{Komponente}{-}{agiere()}
805 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
806 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
807
808 \umlVHVreal{KonkreteKomponente}{Komponente}
809 \umlVHVreal{Dekorierer}{Komponente}
810
811 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
812 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
813
814 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
815 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
816
817 \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
818 \footcite{wiki:dekorierer}
819 \end{tikzpicture}
820 }

```

\liEntwurfsDekoriererCode

```

821 \def\liEntwurfsDekoriererCode{
822 \li@EntwurfsCode{dekorierer}{Komponente}
823 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
824 \li@EntwurfsCode{dekorierer}{Dekorierer}
825 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
826 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
827 \li@EntwurfsCode{dekorierer}{Klient}
828 }

```

\liEntwurfsDekorierer

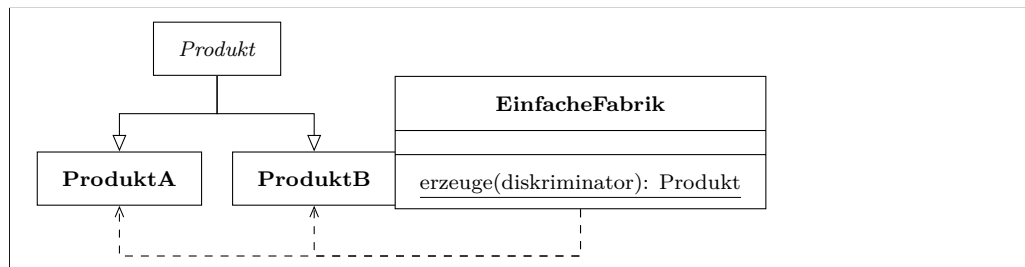
```

829 \def\liEntwurfsDekorierer{
830   \liEntwurfsDekoriererUml
831   \liEntwurfsDekoriererAkteure
832   \liEntwurfsDekoriererCode
833 }

```

## 2.10.7 Einfache Fabrik (Simple Factory)

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



```

834 \def\liEntwurfsEinfacheFabrikUml{
835   \begin{tikzpicture}
836     \umlsimpleclass[type=abstract]{Produkt}
837     \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
838     \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
839     \umlVHVinherit{Produkt}{ProduktA}
840     \umlVHVinherit{Produkt}{ProduktB}
841     \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
842     }{
843       \umlstatic{erzeuge(diskriminator): Produkt}\\
844     }
845     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
846     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
847   \end{tikzpicture}
848 }

```

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.

```

849 \def\liEntwurfsEinfacheFabrikAkteure{
850   \begin{description}
851     \item[EinfacheFabrik]
852
853     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
854     Bedingung verschiedene Objekt instanziiert.
855
856     \item[Produkt]
857
858     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
859
860     \item[KonkretesProdukt]
861
862     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
863   \end{description}
864 }

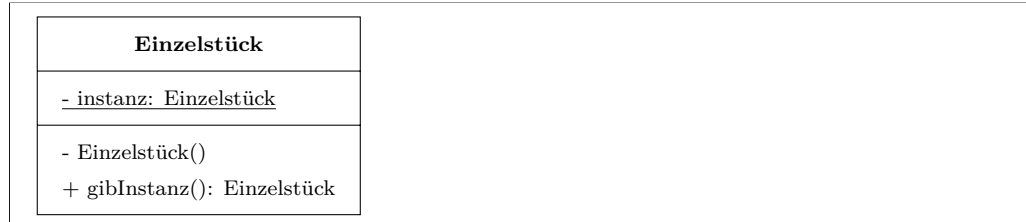
```

\liEntwurfsEinfacheFabrik

```
865 \def\liEntwurfsEinfacheFabrik{
866   \liEntwurfsEinfacheFabrikUml
867   \liEntwurfsEinfacheFabrikAkteure
868 }
```

### 2.10.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```
869 \def\liEntwurfsEinzelstueckUml{
870   \begin{tikzpicture}
871     \umlclass{Einzelstück}{
872       \umlstatic{- instanz: Einzelstück}\\
873     }{
874       - Einzelstück()\\
875       + gibInstanz(): Einzelstück
876     }
877   \end{tikzpicture}
878 }
```

iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

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

```
879 \def\liEntwurfsEinzelstueckAkteure{
880   \begin{description}
881     \item[Einzelstück (Singleton)]
882
883     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
884     nur auf eine einzige Instanz der Klasse zugreifen können.
885   \end{description}
886 }
```

\liEntwurfsEinzelstueckCode

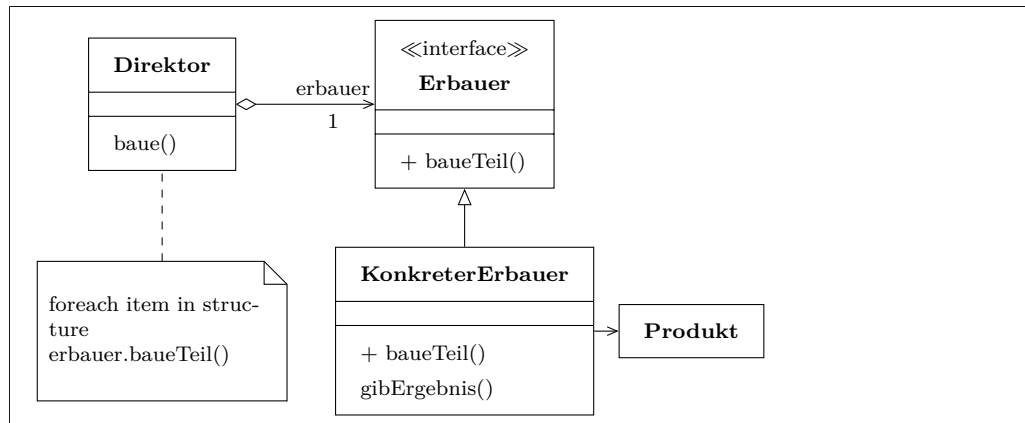
```
887 \def\liEntwurfsEinzelstueckCode{
888   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
889 }
```

\liEntwurfsEinzelstueck

```
890 \def\liEntwurfsEinzelstueck{
891   \liEntwurfsEinzelstueckUml
892   \liEntwurfsEinzelstueckAkteure
893   \liEntwurfsEinzelstueckCode
894 }
```

### 2.10.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

895 \def\liEntwurfsErbauerUml{
896   \begin{tikzpicture}
897     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
898     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
899     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
900       + baueTeil()\
901       gibErgebnis()}
902     \umlclass[x=0,y=3]{Direktor}{}{baue()}
903
904     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
905     \umluniassoc{KonkreterErbauer}{Produkt}
906     \umlinherit{KonkreterErbauer}{Erbauer}
907
908     \umlnote[x=0,y=0,width=3cm]{Direktor}{
909       foreach item in structure\
910       erbauer.baueTeil()
911     }
912   \end{tikzpicture}
913   \footcite{wiki:erbauer}
914 }

```

\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.

```

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

```

```

926     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
927     Schnittstelle zum Auslesen des Produkts.
928
929     \item[Direktor]
930
931     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
932     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
933     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
934     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
935     Klienten.
936
937     \item[Produkt]
938
939     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
940     \footcite{wiki:erbauer}
941 \end{description}
942 }

```

\liEntwurfsErbauer

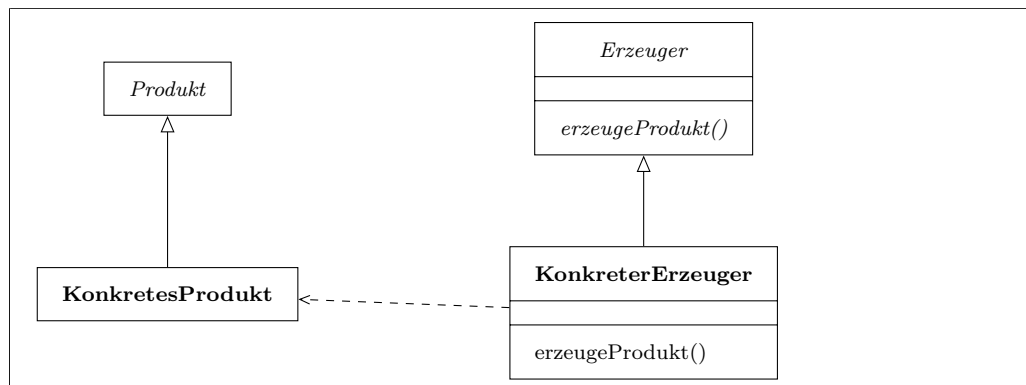
```

943 \def\liEntwurfsErbauer{
944     \liEntwurfsErbauerUml
945     \liEntwurfsErbauerAkteure
946 }

```

## 2.10.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

947 \def\liEntwurfsFabrikmethodeUml{
948     \begin{tikzpicture}
949         \umlsimpleclass[type=abstract]{Produkt}
950         \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
951         \umlinherit{KonkretesProdukt}{Produkt}
952
953         \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
954             \textit{erzeugeProdukt()}\}
955         }
956         \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
957             erzeugeProdukt()
958         }
959         \umlinherit{KonkreterErzeuger}{Erzeuger}
960
961         \umldep{KonkreterErzeuger}{KonkretesProdukt}
962     \end{tikzpicture}
963 }

```

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).

```

964 \def\liEntwurfsFabrikmethodeAkteure{
965   \begin{description}
966     \item[Produkt]
967
968     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
969     zu erzeugende Produkt.
970
971     \item[KonkretesProdukt]
972
973     KonkretesProdukt implementiert die Produkt-Schnittstelle.
974
975     \item[Erzeuger]
976
977     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
978     zu erzeugen und kann eine Default-Implementierung beinhalten.
979
980     \item[KonkreterErzeuger]
981
982     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
983     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
984     Konstruktor einer konkreten Produkt-Klasse aufruft).
985
986     \footcite{wiki:fabrikmethode}
987   \end{description}
988 }

```

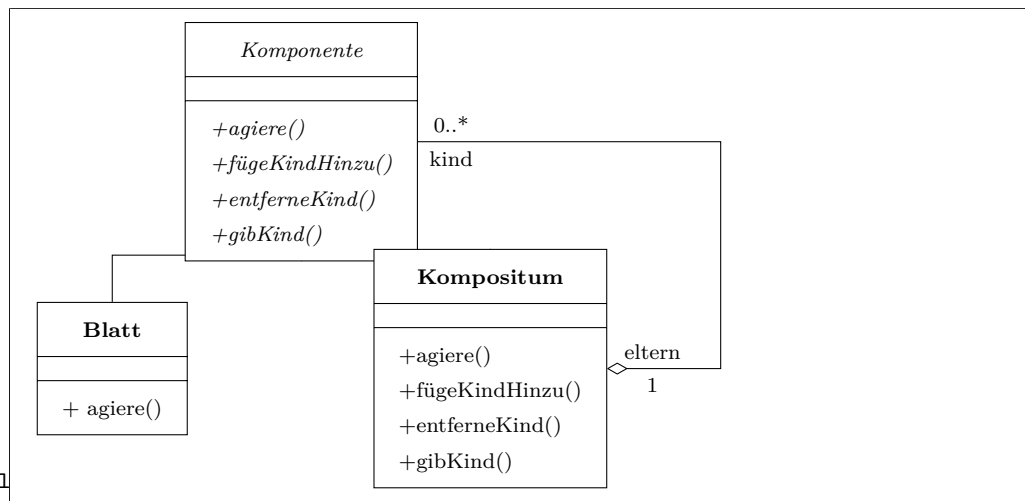
\liEntwurfsFabrikmethode

```

989 \def\liEntwurfsFabrikmethode{
990   \liEntwurfsFabrikmethodeUml
991   \liEntwurfsFabrikmethodeAkteure
992 }

```

### 2.10.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

993 \def\liEntwurfsKompositumUml{
994   \begin{tikzpicture}
995     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
996       \textit{+agiere()}\
997       \textit{+fügeKindHinzu()}\
998       \textit{+entferneKind()}\
999       \textit{+gibKind()}
1000   }
1001   \umlclass[x=0]{Blatt}{-}{+ agiere()}
1002   \umlclass[x=5]{Kompositum}{-}{-}{
1003     +agiere()\
1004     +fügeKindHinzu()\
1005     +entferneKind()\
1006     +gibKind()
1007   }
1008
1009   \umlVHVinherit{Kompositum}{Komponente}
1010   \umlVHVinherit{Blatt}{Komponente}
1011   \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1012   \end{tikzpicture}
1013 }

```

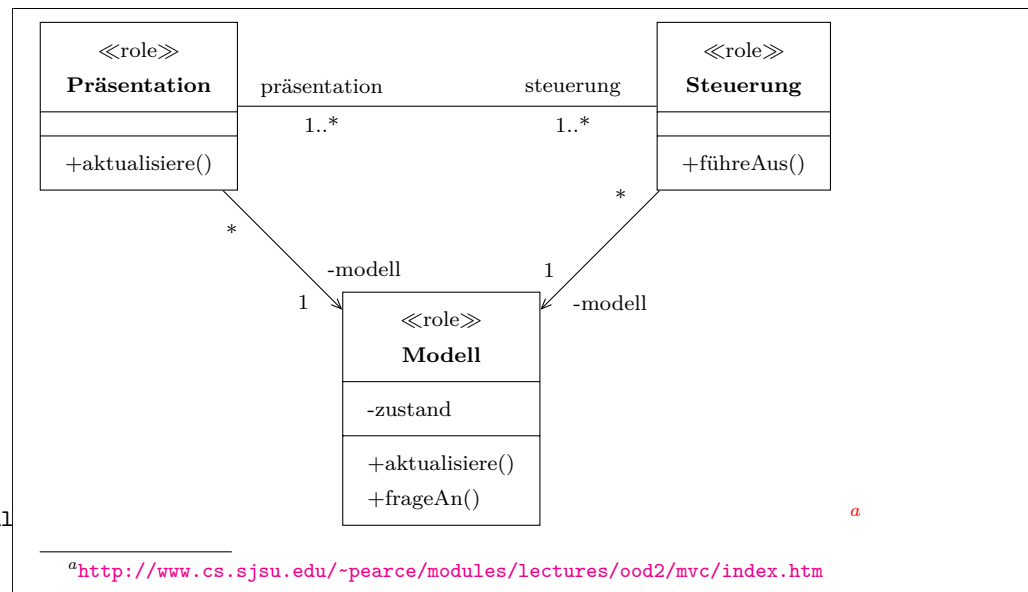
\liEntwurfsFabrikmethode

```

1014 \def\liEntwurfsKompositum{
1015   \liEntwurfsKompositumUml
1016   \liEntwurfsKompositumAkteure
1017 }

```

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



```

1018 \def\liEntwurfsModellPraesentationSteuerungUml{
1019   \begin{tikzpicture}
1020     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1021     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1022     \umlclass[y=-4cm,type=role]{Modell}{-}{
1023       -zustand
1024     }{
1025       +aktualisiere()\
1026       +frageAn()
1027     }
1028
1029     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1030     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1031 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1032 \end{tikzpicture}
1033 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1034 }

```

ModellPraesentationSteuerung

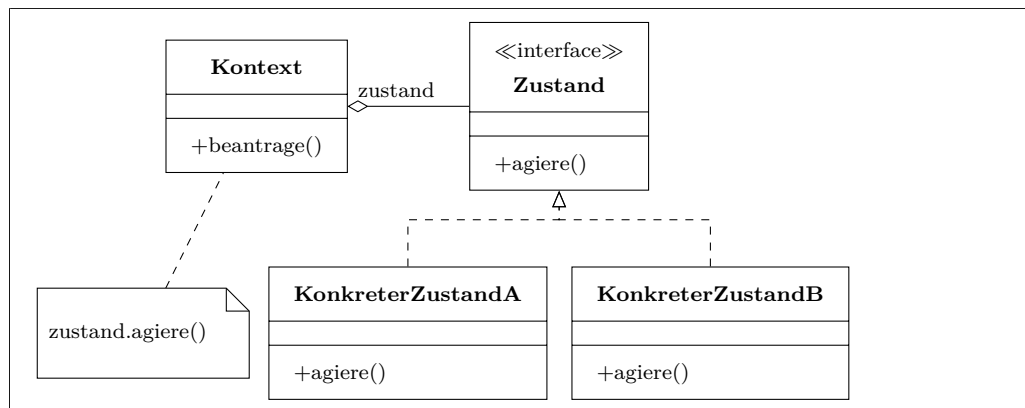
```

1035 \def\liEntwurfs{
1036 \liEntwurfsModellPraesentationSteuerungUml
1037 \liEntwurfsModellPraesentationSteuerungAkteure
1038 }

```

### 2.10.13 Zustand (State)

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



```

1039 \def\liEntwurfsZustandUml{
1040 \begin{tikzpicture}
1041 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1042 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1043 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1044 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1045
1046 \umlVHVreal{KonkreterZustandA}{Zustand}
1047 \umlVHVreal{KonkreterZustandB}{Zustand}
1048
1049 \umlagg[reg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1050
1051 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1052 \end{tikzpicture}
1053 }

```

\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.

```

1054 \def\liEntwurfsZustandAkteure{
1055 \begin{description}
1056 \item[Kontext (Context)]
1057
1058 definiert die clientseitige Schnittstelle und verwaltet die separaten
1059 Zustandsklassen.
1060

```

```

1061 \item[State (Zustand)]
1062
1063 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1064 implementiert gegebenenfalls ein Standardverhalten.
1065
1066 \item[KontreterZustand (ConcreteState)]
1067
1068 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1069 verbunden ist.
1070 \end{description}
1071 }

```

\liEntwurfsZustand

```

1072 \def\liEntwurfsZustand{
1073 \liEntwurfsZustandUml
1074 \liEntwurfsZustandAkteure
1075 }

1076

```

## 2.11 er.sty

```
1077 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1078 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1079 ER-Diagrammen]

1080 \RequirePackage{tikz-er2}
1081 \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}

1082 \RequirePackage{soul}
1083 \RequirePackage{fontawesome}

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

1084 \ExplSyntaxOn

\liErEntity
1085 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
1086 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
1087 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\ e=\liErMpEntity
1088 \def\liErMpEntity#1{
1089   \liErEntity{#1}
1090   \marginpar{
1091     \liErEntity{\tiny\faSquareO{}}~E:~#1}
1092   }
1093 }
□

\liErMpRelationship Let-Abkürzung: \let\ r=\liErMpRelationship
1094 \def\liErMpRelationship#1{
1095   \liErRelationship{#1}
1096   \marginpar{
1097     \liErRelationship{\tiny\faGg{}}~R:~#1}
1098   }
1099 }

\liErMpAttribute Let-Abkürzung: \let\ a=\liErMpAttribute
1100 \def\liErMpAttribute#1{
1101   \liErAttribute{#1}
1102   \marginpar{
1103     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1104   }
1105 }

\liErDatenbankName Let-Abkürzung: \let\ d=\liErDatenbankName
datenbank name
1106 \def\liErDatenbankName#1{
1107   {
1108     \footnotesize\texttt{(#1)}
1109   }
1110 }

1111 \ExplSyntaxOff
1112

```

## 2.12 formale-sprachen.sty

```

1113 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1114 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1115 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1116 \directlua{
1117   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1118 }

1119 \RequirePackage{hyperref}

1120 \liLadePakete{mathe,typographie}

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

1121 \def\liMengeOhneMathe#1{\{ #1 \}}
1122 \def\liMenge#1{%
1123   \ifmmode%
1124     \liMengeOhneMathe{#1}%
1125   \else%
1126     $\liMengeOhneMathe{#1}$%
1127   \fi%
1128 }

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

1129 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1130 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1131 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1132 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1133 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1134 \let\liZustandsmenge=\liPotenzmenge

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

1135 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1136 \def\liUeberfuehrungsFunktion#1{
1137   \ifmmode
1138     \liUeberfuehrungsFunktionOhneMathe{#1}
1139   \else
1140     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1141   \fi
1142 }

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

1143 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1144 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1145 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1146 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1147 \def\zustandsnamens@liste#1#2{
1148   $
1149   \{
1150     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1151   \}
1152   $
1153 }
1154 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1155 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1156 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1157 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1158 \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}

1159 \NewDocumentEnvironment { liProduktionsRegeln }
1160 { 0{P} +b }
1161 {
1162   \liGeschweifteKlammern{#1}
1163   {
1164     \begin{align*}
1165       \directlua{formale_sprachen.produktions_regeln('#2')}
1166       \end{align*}
1167     }{-0.2cm}{-1.5cm}
1168   } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1169 \def\liProduktionen#1{
1170   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1171 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1172 \def\liZustandsnameTiefgestellt#1{
1173   \ifmmode
1174     \liZustandsBuchstabe\sb{#1}
1175   \else
1176     $\liZustandsBuchstabe\sb{#1}$
1177   \fi
1178 }

1179 \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}

```



```

1180 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1181   $
1182   \tl_if_empty:nTF {#1} {} {#1 =}
1183   \{
1184     \, #2 \,
1185     |
1186     \, #3 \,
1187   \}$
1188 }
1189 \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

1190 \def\liFlaci#1{%
1191   \par
1192   {%
1193     \scriptsize
1194     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1195     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1196     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1197     \href{https://flaci.com/#1}{flaci.com/#1}%
1198   }%
1199   \par
1200 }

\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}:  $G = (V, \Sigma, P, X)$ 

1201 \ExplSyntaxOn
1202 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1203   \tl_set:Nn \l_variablen_tl {V}
1204   \tl_set:Nn \l_alphabet_tl {\Sigma}
1205   \tl_set:Nn \l_produktionen_tl {P}
1206   \tl_set:Nn \l_start_tl {S}
1207
1208   \keys_define:nn { grammatik } {
1209     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1210     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1211     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1212     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1213   }
1214
1215   \keys_set:nn { grammatik } { #2 }
1216
1217   $#1 = (
1218     \l_variablen_tl,
1219     \l_alphabet_tl,
1220     \l_produktionen_tl,
1221     \l_start_tl
1222   )$
1223 }
1224 \ExplSyntaxOff

1225

```

## 2.13 formatierung.sty

```
1226 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1227 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

### 2.13.1 Schriftarten / Typographie

```
1228 \RequirePackage{mathpazo}
1229 \RequirePackage[no-math]{fontspec}
1230 \setmainfont{texgyrepagella}
1231 \setsansfont{QTAncientOlive}
1232 \RequirePackage{sectsty}
1233 \allsectionsfont{\fontspec{QTAncientOlive}}
```

### 2.13.2 Farben

```
1234 \RequirePackage{xcolor}
1235 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

### 2.13.3 Überschriften

```
1236 \RequirePackage{titlesec}
1237 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1238 \titlespacing{\chapter}{0pt}{0pt}{*1}
1239 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1240 \setcounter{secnumdepth}{0}
```

### 2.13.4 Listen

```
1241 \RequirePackage{paralist}
1242 \renewcommand\labelitemi{-}
1243 \renewcommand\labelitemii{-}
1244 \renewcommand\labelitemiii{-}
1245 \renewcommand\labelitemiv{-}
1246 % Counter: enumi enumii enumiii enumiv
1247 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1248 \renewcommand{\labelenumi}{(\alph{enumi})}
1249 \renewcommand{\labelenumii}{(\roman{enumii})}
```

### 2.13.5 Kasten

```
1250 \RequirePackage{mdframed}
1251 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1252 \NewDocumentEnvironment { liKasten }{ } {
1253   \begin{mdframed}[backgroundcolor=white!0]
1254 } {
1255   \end{mdframed}
1256 }
```

### 2.13.6 Header

```
1257 \RequirePackage{fancyhdr}
1258 \fancyhead[L,C,R]{}
1259 \fancyfoot[L]{}
1260 \fancyfoot[C]{}
1261 \fancyfoot[R]{\thepage}
1262 \pagestyle{fancy}
1263 \renewcommand{\headrulewidth}{0pt}
1264 \renewcommand{\footrulewidth}{0pt}
1265
```

## 2.14 gantt.sty

```

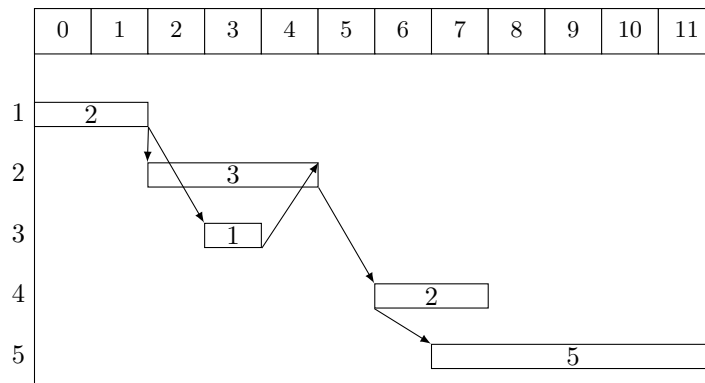
1266 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1267 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1268 \RequirePackage{tikz-uml}
1269 \RequirePackage{pgfgantt}
1270 \setganttlinklabel{f-s}{}
1271 \setganttlinklabel{s-s}{}
1272 \setganttlinklabel{f-f}{}
1273 \setganttlinklabel{s-f}{}

1274

```

## 2.15 grafik.sty

```
1275 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1276 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1277 TikZ, um Grafiken zeichnen zu können.]
1278 \RequirePackage{tikz}
1279
```

## 2.16 graph.sty

```

1280 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1281 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1282 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightharrow`)

```

1283 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

$$\begin{array}{c}
 \\
 a \\
 b \\
 c \\
 d \\
 e
 \end{array}
 \begin{pmatrix}
 & a & b & c & d & e \\
 \begin{pmatrix}
 0 & 1 & 0 & 4 & 0 \\
 0 & 0 & 0 & 1 & 3 \\
 0 & 0 & 0 & 1 & 0 \\
 4 & 1 & 1 & 0 & 2 \\
 0 & 0 & 0 & 0 & 0
 \end{pmatrix}
 \end{pmatrix}$$

```

1284 \RequirePackage{blkarray}
1285 \usetikzlibrary{arrows.meta}

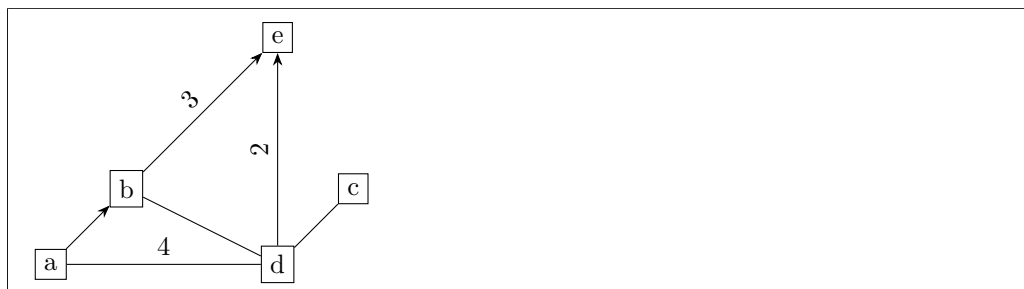
```

```

\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}

```



```

1286 \tikzset{
1287   li graph/.style={
1288     every node/.style={
1289       rectangle,
1290       draw,
1291     },
1292     every edge/.style={
1293       >={Stealth[black]},
1294       draw,
1295     },
1296     every edge/.append style={
1297       every node/.style={
1298         sloped,
1299         auto,
1300       }
1301     }
1302   },
1303   li markierung/.style={
1304     ultra thick,
1305   }
1306 }

```

**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}

```

```

1307 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1308

```

## 2.17 hanoi.sty

```
1309 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1310 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1311 von Hanoi-Grafiken]
```

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

```
1312 \RequirePackage{tikz}
1313 \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}

1314 \def\li@mset #1[#2]=#3{%
1315 \expandafter\xdef\csname #1#2\endcsname{#3}
1316 }
1317 \def\li@mget #1[#2]{%
1318 \csname #1#2\endcsname
1319 }
1320 \def\li@minc #1[#2]+=#3{%
1321 \pgfmathparse{\li@mget #1[#2]+#3}%
1322 \li@mset #1[#2]=\pgfmathresult
1323 }
1324
1325 \def\liHanoi#1#2{
1326   \edef\li@numdiscs{#1}
1327   \def\li@sequence{#2}
1328   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1329     % init colors
1330     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1331       \li@mset col[\j]={\c};
1332     }
1333     % draw poles and init pole counters
1334     \foreach \j in {1,2,3}{
1335       \li@mset pos[\j]=0
1336       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1337     }
1338     % draw base
1339     \draw (.5,-.5) -- +(3,0);
1340     % draw discs
1341     \foreach[count=\k] \i/\j in \li@sequence{
1342       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1343       \li@minc pos[\j]+=.5
1344     }
1345   \end{tikzpicture}
1346 }
```

## 2.18 komplexitaetstheorie.sty

```

1347 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1348 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1349 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1350 Polynomialzeitreduktion.]

```

### Let-Abkürzungen

```

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

```

```

1351 \liLadePakete{mathe}

```

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

```

1352 \RequirePackage{mdframed}

```

```

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

```

```

1353 \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`

```

1354 \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`

```

1355 \def\liProblemBeschreibung#1#2#3{
1356   \begin{mdframed}[
1357     userdefinedwidth=9cm,
1358     align=center,
1359     backgroundcolor=white!0,
1360   ]
1361     \centerline{\large\liProblemName{#1}}
1362
1363     \medskip
1364
1365     \begin{description}
1366       \item[Gegeben:] #2
1367       \item[Frage:] #3
1368     \end{description}
1369   \end{mdframed}
1370 }

```



```

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

1371 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1372 \begin{displaymath}
1373 \liProblemName{#1}
1374 \preceq_{#2}
1375 \liProblemName{#3}
1376 \end{displaymath}
1377 }

\liProblemVertexCover

1378 \def\liProblemClique{%
1379 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1380 Mindestgröße  $n$  in einem gegebenen Graphen.
1381 \footcite[Seite 76]{theo:fs:4}
1382 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1383 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1384 \footcite{wiki:cliquenproblem}
1385 }

\liProblemVertexCover

1386 \def\liProblemVertexCover{%
1387 %
1388 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1389 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1390 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1391 \footcite{wiki:knotenueberdeckung}
1392
1393 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1394 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1395 mindestens einem Knoten aus  $U$  verbunden ist.
1396 \footcite[Seite 78]{theo:fs:4}%
1397 }

\liProblemSubsetSum Kein Karp-21-Problem

1398 \def\liProblemSubsetSum{%
1399 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1400 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1401 \footcite{wiki:teilsommenproblem}
1402 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1403 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1404 größer als eine gegebene obere Schranke  $c$  ist.
1405 \footcite[Seite 74]{theo:fs:4}
1406 }

\liProblemSubsetSum Kein Karp-21-Problem

1407 \def\liProblemSat{%
1408 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1409 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1410 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1411 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1412 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1413 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1414 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1415 aufgestellt werden.
1416 \footcite[Seite 71]{theo:fs:4}
1417 }

1418

```

## 2.19 kontrollflussgraph.sty

```
1419 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1420 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1421 \RequirePackage{tikz}
1422 \usetikzlibrary{positioning}
1423 \tikzset{
1424   li kontrollfluss/.style={
1425     knoten/.style={
1426       circle,
1427       draw
1428     },
1429     usebox/.style={
1430       draw,
1431       rectangle,
1432       font=\scriptsize,
1433       anchor=west,
1434       align=left,
1435     },
1436     bedingung/.style={
1437       midway,
1438       draw=none,
1439       font=\scriptsize
1440     },
1441     knotenbeschriftung/.style={
1442       draw,
1443       rectangle,
1444       midway,
1445       font=\scriptsize
1446     },
1447     wahr/.style={
1448       thick
1449     },
1450     falsch/.style={
1451       dashed
1452     },
1453     every node/.style={
1454       circle,
1455       draw,
1456     },
1457     every edge/.append style={
1458       every node/.style={
1459         draw=none,
1460         bedingung,
1461       }
1462     },
1463     every path/.style={
1464       draw,
1465       ->,
1466     },
1467     every pin/.style={
1468       draw,
1469       dotted,
1470       rectangle,
1471       pin position=right
1472     },
1473     every pin edge/.style={
1474       dotted,
1475       arrows=-,
1476     }
1477   }
1478 }
```

liKontrollflussgraph

```

1479 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1480   \begin{tikzpicture}[
1481     li kontrollfluss,
1482     #1
1483   ]
1484 } {
1485   \end{tikzpicture}
1486 }

\liAnweisung
1487 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1488 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1489 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1490 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1491 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1492 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1493 \ExplSyntaxOn
1494 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1495 {
1496   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1497   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1498   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1499 }
1500 \ExplSyntaxOff

1501

```

## 2.20 kopf-fusszeilen.sty

```
1502 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1503 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1504 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1505 \ExplSyntaxOn

1506 \fancyhead{}
1507 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1508 \fancyhead[L0,RE]{\scriptsize\today}
1509 \fancyfoot{}
1510 \fancyfoot[LE,R0]{\thepage}
1511 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1512 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1513 \renewcommand{\headrulewidth}{0.4pt}
1514 \renewcommand{\footrulewidth}{0.4pt}

1515 \ExplSyntaxOff

1516
```

## 2.21 literatur-dummy.sty

```
1517 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1518 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1519 \def\literatur{}

\footcite

1520 % \RequirePackage[stable,multiple]{footmisc}
1521 \NewDocumentCommand{ \footcite } { o m }{}

1522
```

## 2.22 literatur.sty

```
1523 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1524 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1525 \RequirePackage{csquotes}
1526 \RequirePackage[
1527   bibencoding=utf8,
1528   citestyle=authortitle,
1529   backend=biber,
1530 ]{biblatex}
1531 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1532 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1533 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1534 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1535 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1536 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1537 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1538 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1539 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1540 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1541 % To allow footnotes in the heading
1542 \RequirePackage[stable,multiple]{footmisc}

\literatur

1543 \def\literatur{\printbibliography[heading=subbibliography]}

1544
```

## 2.23 makros.sty

```

1545 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1546 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1547 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1548 anderen Paket passen]
1549 \RequirePackage{hyperref}
1550 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1551 \RequirePackage{paralist}

\inhaltsverzeichnis
1552 \def\inhaltsverzeichnis {
1553   \begin{mdframed}
1554     \begin{group}
1555       \let\clearpage\relax
1556       \tableofcontents
1557     \end{group}
1558   \end{mdframed}
1559 }

\memph \mephm (\marginpar and \emph)
1560 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1561 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1562 \newcommand{\liPseudoUeberschrift}[1]{
1563   \bigskip
1564   \noindent
1565   \textsf{\textbf{#1}}
1566   \noindent
1567 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1568 \newcommand{\liBeschriftung}[1]{
1569   \par
1570   \noindent
1571   \medskip
1572   \textbf{#1}:
1573   \medskip
1574   \noindent
1575 }

\hinweis
1576 \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.
1577 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1578 \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.
1579 \RequirePackage{xparse}
1580 \ExplSyntaxOn

```

```

1581 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1582 {
1583   \str_case:nn {#1} {
1584     {standard} {
1585       \def\beschriftung{}
1586       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1587     }
1588     {richtig} {
1589       \def\beschriftung{richtig}
1590       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1591     }
1592     {falsch} {
1593       \def\beschriftung{falsch}
1594       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1595     }
1596     {muster} {
1597       \def\beschriftung{Musterlösung}
1598       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1599     }
1600   }
1601   \ifx\beschriftung\empty\else
1602     \noindent
1603     \textbf{\beschriftung{:}}
1604     \fi
1605     \begin{mdframed}
1606   }
1607 {\end{mdframed}}

```

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

```

1608 \NewDocumentEnvironment{ liAdditum } { o }
1609 {
1610   \begin{mdframed}[backgroundcolor=yellow!5]
1611     \IfNoValueTF {#1}
1612     { \liPseudoUeberschrift{Additum} }
1613     { \liPseudoUeberschrift{Additum:~#1} }
1614   }
1615 {\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.

```

1616 \NewDocumentEnvironment{ liExkurs }{o +b}{
1617   \vspace{0.2cm}%
1618   \begin{mdframed}[
1619     backgroundcolor=white,
1620     bottomline=false,
1621     innermargin=1cm,
1622     leftline=true,
1623     linecolor=black,
1624     linewidth=0.1cm,
1625     outermargin=1cm,
1626     rightline=false,
1627     topline=false,
1628   ]

```



```

1629 \footnotesize
1630 \noindent%
1631 \textbf{Exkurs:~#1}\par%
1632 \noindent%
1633 #2
1634 \end{mdframed}
1635 \vspace{0.2cm}
1636 }{}

```

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

```

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

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

1637 \cs_new:Npn \listen@punkt #1 {\item #1}
1638 \NewDocumentEnvironment { liQuellen }{ +b }
1639 {
1640   \seq_clear_new:N \l_quellen
1641   \seq_set_split:Nnn \l_quellen {\item} {#1}
1642   \seq_remove_all:Nn \l_quellen {}
1643   \begin{mdframed}[backgroundcolor=white!0]
1644     \footnotesize
1645     \noindent
1646     \textsf{\textbf{Weiterführende~Literatur:}}
1647     \medskip
1648     \begin{compactitem}
1649       \seq_map_function:NN \l_quellen {\listen@punkt}
1650     \end{compactitem}
1651   \end{mdframed}
1652   %
1653   \makeatletter
1654   \par\@afterindentfalse\@afterheading
1655   \makeatother
1656 } {}

```

liLernkartei

```

1657 \NewDocumentEnvironment { liLernkartei }{ m +b }
1658 {
1659   \begin{mdframed}
1660     \footnotesize
1661     \noindent%
1662     \textbf{Lernkarteikarte:~#1}\par%
1663     \noindent%
1664     #2
1665   \end{mdframed}
1666 } {}

```

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

```

1667 \NewDocumentEnvironment { liDiagramm }{ m +b }
1668 {
1669   \begin{mdframed}[backgroundcolor=white!0]
1670     \small
1671     \noindent%
1672     \textit{#1}:
1673     \begin{center}

```

```

1674 #2
1675 \medskip
1676 \end{center}
1677 \end{mdframed}
1678 } {}
1679 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1680 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1681 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1682 }
1683

\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.
1684 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1685 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1686 }

\zB
1687 \def\zB{z.\.,B. }

\ZB
1688 \def\ZB{Z.\.,B. }

\dh
1689 \def\dh{d.\.,h. }

1690

```

## 2.24 master-theorem.sty

1691 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1692 \ProvidesPackage{lehramt-informatik-master-theorem}[2021/04/13]

### Let-Abkürzungen

\let\0=\li0

\let\o=\li0omega

\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}

1693 \ExplSyntaxOn

1694 \RequirePackage{amsmath}

\liRundeKlammer

1695 \def\liRundeKlammer#1{

1696 \negthinspace \left( #1 \right)

1697 }

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

1698 \def\liThetaOhneMathe#1{

1699 \Theta \liRundeKlammer{#1}

1700 }

1701 \def\liTheta#1{

1702 \ifmmode

1703 \liThetaOhneMathe{#1}

1704 \else

1705 \$\liThetaOhneMathe{#1}\$

1706 \fi

1707 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1708 \def\liOmegaOhneMathe#1{
1709 \Omega \liRundeKlammer{#1}
1710 }
1711 \def\liOmega#1{
1712 \ifmmode
1713 \liOmegaOhneMathe{#1}
1714 \else
1715 $\liOmegaOhneMathe{#1}$
1716 \fi
1717 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1718 \def\liOOhneMathe#1{
1719 \mathcal{O} \liRundeKlammer{#1}
1720 }
1721 \def\liO#1{
1722 \ifmmode
1723 \liOOhneMathe{#1}
1724 \else
1725 $\liOOhneMathe{#1}$
1726 \fi
1727 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1728 \def\liTOhneMathe#1#2{
1729 \tl_if_blank:nTF {#1}
1730 {}
1731 {#1 \cdot }
1732 T
1733 \liRundeKlammer{\frac{n}{#2}}
1734 }
1735 \def\liT#1#2{
1736 \ifmmode
1737 \liTOhneMathe{#1}{#2}
1738 \else
1739 $\liTOhneMathe{#1}{#2}$
1740 \fi
1741 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1742 \def\liRekursionsGleichung{
1743 $T(n) = \liT{a}{b} + f(n)$
1744 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1745 \def\liBedingungEins{
1746 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1747 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1748 \def\liBedingungZwei{
1749 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1750 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1751 \def\liBedingungDrei{
1752 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1753 }

1754 \ExplSyntaxOff

```

## \liMasterVariablen

```

1755 \def\liMasterVariablen{
1756   \begin{displaymath}
1757     T(n) = \liT{a}{b} + f(n)
1758   \end{displaymath}
1759
1760   \begin{itemize}
1761     \item[$a = $]
1762       Anzahl der Unterprobleme in der Rekursion
1763
1764     \item[$\textstyle{\frac{1}{b}} = $]
1765       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1766       repräsentiert wird
1767
1768     \item[$f(n) = $]
1769       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1770       die Kombination der Teillösungen entstehen
1771   \end{itemize}
1772   \footcite{wiki:master-theorem}
1773   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1774 }

```

## \liMasterFaelle

```

1775 \def\liMasterFaelle{
1776   \begin{description}
1777     \item[1. Fall:]
1778        $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1779
1780       \hfill falls \liBedingungEins
1781       für  $\varepsilon > 0$ 
1782
1783     \item[2. Fall:]
1784        $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1785
1786       \hfill falls \liBedingungZwei
1787
1788     \item[3. Fall:]
1789        $T(n) \in \liTheta{f(n)}$ 
1790
1791       \hfill falls \liBedingungDrei
1792       für  $\varepsilon > 0$ 
1793       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1794       gilt:
1795        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1796     \end{description}
1797 }

```

## \liMasterVariablenDeklaration

```

1798 \def\liMasterVariablenDeklaration#1#2#3{
1799   \begin{description}
1800     \item[Allgemeine Rekursionsgleichung:] \strut
1801
1802     \liRekursionsGleichung
1803
1804     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1805
1806     #1
1807
1808     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1809
1810     um  $\frac{1}{#2}$  also  $b = #2$ 
1811
1812     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut

```

```

1813
1814     $#3$
1815
1816     \item[Ergibt folgende Rekursionsgleichung:] \strut
1817
1818      $T(n) = T_{\#1\#2} + \#3$ 
1819     \end{description}
1820 }

\liMasterFallRechnung
1821 \def\liMasterFallRechnung#1#2#3{
1822     \begin{description}
1823     \item[1. Fall:] \liBedingungEins:
1824
1825     #1
1826
1827     \item[2. Fall:] \liBedingungZwei:
1828
1829     #2
1830
1831     \item[3. Fall:] \liBedingungDrei:
1832
1833     #3
1834     \end{description}
1835 }

\liMasterExkurs
1836 \def\liMasterExkurs{
1837     \begin{liExkurs}[Master-Theorem]
1838     \liMasterVariablen
1839
1840     \noindent
1841     Dann gilt:
1842
1843     \liMasterFaelle
1844     \end{liExkurs}
1845 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1846 \def\liMasterWolframLink#1{
1847     Berechne die Rekursionsgleichung auf WolframAlpha:
1848     \href{https://www.wolframalpha.com/input/?i=\#1\{WolframAlpha\}}
1849 }

1850

```

## 2.25 mathe.sty

```
1851 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1852 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1853
1854 % for example \ltimes \rtimes
1855 %\RequirePackage{amssymb}
1856 \RequirePackage{amsmath}
1857
1858 %%
1859 % \mlq \mrq
1860 %%
1861 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1862 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1863
```

## 2.26 minimierung.sty

```

1864 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1865 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1866 Minimierungsalgorithmus von einem NEA zu einem DEA]

1867 \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

1868 \def\liFussnote#1{\$x_{#1}\$}

1869 \def\li@fussnote@text#1#2{
1870 \liFussnote{#1}
1871 \quad
1872 {\footnotesize #2}
1873 }

\liFussnoteEinsText

1874 \def\liFussnoteEinsText{
1875 \li@fussnote@text{1}
1876 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1877 }

\liFussnoteZweiText

1878 \def\liFussnoteZweiText{
1879 \li@fussnote@text{2}
1880 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1881 }

\liFussnoteDreiText

1882 \def\liFussnoteDreiText{
1883 \li@fussnote@text{3}

```



```

1884 {In weiteren Iterationen markierte Zustände.}
1885 }

\liFussnoteVierText
1886 \def\liFussnoteVierText{
1887   \li@fussnote@text{4}
1888   {...}
1889 }

\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.                          |
| $x_4$ | ...                                                                  |



1890 \def\liFussnoten{
1891   \bigskip
1892
1893   \noindent
1894   \liFussnoteEinsText
1895
1896   \noindent
1897   \liFussnoteZweiText
1898
1899   \noindent
1900   \liFussnoteDreiText
1901
1902   \noindent
1903   \liFussnoteVierText
1904 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1905 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1906 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1907 \def\liZustandsPaar#1#2{
1908   $(
1909     \liZustandsPaarVariablenName_#1,
1910     \liZustandsPaarVariablenName_#2
1911   )$
1912 }

liUebergangsTabelle
1913 \renewcommand{\arraystretch}{1.4}
1914 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1915   \liPseudoUeberschrift{Übergangstabelle}
1916   \begin{center}
1917     \begin{tabular}{r|l|l}
1918       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
1919     \end{tabular}
1920   \end{center}
1921 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1923 \ExplSyntaxOn
1924 \def\liUeberschriftDreiecksTabelle{
1925   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1926 }

```

**\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. —

```

1927 \def\liMinimierungErklaerung{
1928   %\footcite[Seite~19]{koenig}
1929   \liParagraphMitLinien{
1930     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1931     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1932     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1933     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1934      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1935     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1936     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1937     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1938     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1939   }
1940 }
1941 \ExplSyntaxOff
1942

```

## 2.27 normalformen.sty

```
1943 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1944 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
1945 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1946 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
1947 \liLadePakete{mathe,typographie}
```

```
1948 \directlua{
1949   helper = require('lehramt-informatik-helfer')
1950   normalformen = require('lehramt-informatik-normalformen')
1951 }
```

### 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
```

```
1952 \def\liTeilen#1{
1953   \directlua{tex.print(normalformen.teilen('#1'))}
1954 }
```

```
\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.)*\)\ \ah{$1}
1955 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
1956 \def\liAttributHuelle#1{
1957   \ifmmode
1958     \liAttributHuelleOhneMathe{\#1}
1959   \else
1960     $\liAttributHuelleOhneMathe{\#1}$
1961   \fi
1962 }
```

```
\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
1963 \def\liAttributMenge#1{\{ \textit{\#1} \}}
```

liAHuelle

```
1964 \NewDocumentEnvironment{ liAHuelle } { +b } {
1965   \begin{group}
1966   \footnotesize
1967   \begin{multline*}
1968     \#1
1969   \end{multline*}
1970   \end{group}
1971 } { }
```

```
\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
1972 \def\liLinksReduktion#1#2#3{
1973   \shoveleft{
1974     \liAttributHuelleOhneMathe{FA,
1975       \liAttributMenge{\#1 \string\ #2}} =
1976     } \}
```

```

1977 \shoveright{
1978 \liAttributMenge{#3}
1979 } \\\
1980 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
1981 \def\liLinksReduktionInline#1#2#3{%
1982 {%
1983 \footnotesize%
1984 $\liAttributHuelleOhneMathe{F,
1985 \liAttributMenge{#1 \string\ #2}} =
1986 \liAttributMenge{#3}$
1987 }
1988 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
\ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
1989 \def\liRechtsReduktionInline#1#2#3#4{%
1990 {%
1991 \footnotesize%
1992 $\liAttributHuelleOhneMathe{
1993 F \setminus
1994 \liFunktionaleAbhaengigkeit{#1}
1995 \def\tmp{#2}\ifx\tmp\empty
1996 \else
1997 \cup \liFunktionaleAbhaengigkeit{#2}
1998 \fi
1999 ,
2000 \liAttributMenge{#3}
2001 } =
2002 \liAttributMenge{#4}$
2003 }
2004 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
\fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
\$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2005 \def\liFunktionaleAbhaengigkeit#1{%
2006 \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2007 }

```

```

\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 (.*)\$ \fa{(.*) \\\rightarrow
$1 -> $2;

```

```

2008 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2009   \liGeschweifteKlammern
2010   {#1}
2011   {
2012     \begin{align*}
2013       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2014     \end{align*}
2015   }
2016   {-0.5cm}
2017   {-1.7cm}
2018 }

\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}

2019 \NewDocumentCommand {\liRelation} { O{R} m } {
2020   $\directlua{
2021     local name = helper.konvertiere_tiefgestellt('#1')
2022     tex.print(name)
2023   }$(\textit{\,#2\,})
2024 }

2025

```

## 2.28 petri.sty

2026 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2027 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### Let-Abkürzungen

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2028 \RequirePackage{tikz}

2029 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2030 \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

```
2031 \def\liPetriSetzeSchluessel{%
2032   \def\TmpTransitionOne{}%
2033   \def\TmpTransitionTwo{}%
2034   \def\TmpTransitionThree{}%
2035   \def\TmpTransitionFour{}%
2036   \def\TmpTransitionFive{}%
2037   \def\TmpTransitionSix{}%
2038   \def\TmpTransitionSeven{}%
2039   \def\TmpTransitionEight{}%
2040   \def\TmpTransitionNine{}%
2041   \def\TmpTransitionTen{}%
2042   \pgfkeys{/petri/.cd,
2043     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2044     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2045     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2046     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
2047     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

2048     p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2049     p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2050     p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2051     p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2052     p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2053     t1/.store in=\TmpTransitionOne,t1/.default=activated,
2054     t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2055     t3/.store in=\TmpTransitionThree,t3/.default=activated,
2056     t4/.store in=\TmpTransitionFour,t4/.default=activated,
2057     t5/.store in=\TmpTransitionFive,t5/.default=activated,
2058     t6/.store in=\TmpTransitionSix,t6/.default=activated,
2059     t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2060     t8/.store in=\TmpTransitionEight,t8/.default=activated,
2061     t9/.store in=\TmpTransitionNine,t9/.default=activated,
2062     t10/.store in=\TmpTransitionTen,t10/.default=activated,
2063     scale/.store in=\TmpScale,scale/.default=0.5,
2064     x/.store in=\TmpX,x/.default=5,
2065     y/.store in=\TmpY,y/.default=5,
2066   }%
2067 }

2068 \tikzset{
2069   li petri/.style={
2070     activated/.style={
2071       very thick
2072     },
2073     inhibitor/.style={
2074       {Circle[open,length=2mm,fill=white]}-
2075     }
2076   }
2077 }

```

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

```

2078 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2079 \def\liPetriTransitionsName#1{
2080   \ifmmode
2081     \liPetriTransitionsNameOhneMathe{#1}
2082   \else
2083     $\liPetriTransitionsNameOhneMathe{#1}$
2084   \fi
2085 }

```

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

```

2086 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2087   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2088 }

```

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

```

2089 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2090 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2091 }

```

## 2.29 potenzmengen-konstruktion.sty

```
2092 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2093 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2094 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2095 \liLadePakete{formale-sprachen}
2096 \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}
    }
  }
}
```

```
2097 \def\liZustandsMengenSammlung#1#2{
2098   \liZustandsnameGross{#1}
2099   {
2100     \footnotesize
2101     \liPotenzmenge{
2102       \str_case:nn {#1} {#2
2103       }
2104     }
2105 }
```

```
\liZustandsMengenSammlungNr
```

```
2106 \def\liZustandsMengenSammlungNr#1#2{
2107   \liZustandsnameGross{#1}
2108   {
```



```
2109     \footnotesize
2110     \liZustandsmengeNr{
2111         \str_case:nn {#1} #2
2112     }
2113 }
2114 }

2115 \ExplSyntaxOff
2116
```

## 2.30 pseudo.sty

2117 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2118 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen  
 2119 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>
---

2120 \RequirePackage[german,boxruled]{algorithm2e}

2121

## 2.31 pumping-lemma.sty

2122 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2123 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2124 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2125 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2126 \def\liPumpingRegulaer{%
2127   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2128   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2129    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2130   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2131
2132   \begin{enumerate}
2133     \item  $|v| \geq 1$ 
2134     (Das Wort  $v$  ist nicht leer.)
2135
2136     \item  $|uv| \leq j$ 
2137     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2138
2139     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
2140     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2141     Sprache  $L$ )
2142   \end{enumerate}
2143
2144   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2145   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2146 }
```

\liPumpingKontextfrei

```
2147 \def\liPumpingKontextfrei{%
2148   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2149   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
2150    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2151
2152   \begin{enumerate}
2153     \item  $|vx| \geq 1$ 
2154     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2155
2156     \item  $|vwx| \leq j$ 
2157     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2158
2159     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy \in L$  (Für jede
2160     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2161     Sprache  $L$ )
2162   \end{enumerate}
2163 }
```

2164

## 2.32 quicksort.sty

```

2165 % https://tex.stackexchange.com/a/142634
2166 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2167 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2168
2169 %-----
2170 % USAGE:
2171 % \QSinitialize{comma, separated, numerical, values}
2172 % \loop
2173 % \QSpivotStep
2174 % \ifnum\value{pivotcount}>0
2175 %   \QSSortStep
2176 % \repeat
2177 %-----
2178
2179 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2180 \RequirePackage{xintfrac, xinttools}
2181
2182 \RequirePackage{tikz}
2183
2184 %-----
2185 % FIRST PART: TikZ styles and macros for the actual drawing
2186 \newcounter{cellcount}% used for coordinates of the node
2187 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2188
2189 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2190 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2191 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2192 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2193 % by police of LaTeX good conduct ? )
2194 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2195          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2196          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2197 % this is the "b" style as used in the image below
2198          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2199 % nicer:
2200          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2201          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2202
2203 % NOTE the b style was originally the same as the r(aised) style apart from
2204 % not being raised, but I find it nicer with a somewhat different
2205 % specification. I have not updated the images though.
2206
2207 % How the nodes are drawn depending on whether on the left of the pivot value
2208 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2209
2210 \def\DecoLEFT #1{%
2211   \xintFor* ##1 in {#1} \do
2212     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2213 }
2214
2215 \def\DecoINERT #1{%
2216   \xintFor* ##1 in {#1} \do
2217     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2218 }
2219
2220 \def\DecoRIGHT #1{%
2221   \xintFor* ##1 in {#1} \do
2222     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2223 }
2224
2225 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2226   \xintFor* ##1 in {#1} \do

```

```

2227     {\stepcounter{cellcount}}%
2228     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2229 }
2230
2231 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2232     \xintFor* ##1 in {#1} \do
2233     {\stepcounter{cellcount}}%
2234     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2235 }
2236
2237 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2238     \xintFor* ##1 in {#1} \do
2239     {\stepcounter{cellcount}}%
2240     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2241 }
2242
2243 %-----
2244 % SECOND PART: the actual sorting routines.
2245
2246 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2247 \def\QS@sort@b #1{\ifcase #1
2248     \expandafter\QS@sort@empty
2249     \or\expandafter\QS@sort@single
2250     \else\expandafter\QS@sort@c
2251     \fi
2252 }%
2253 \def\QS@sort@empty #1{}
2254 \def\QS@sort@single #1{\QSIr {#1}}
2255
2256 % This step is to pick the last as pivot.
2257 \def\QS@sort@c #1%
2258     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2259
2260 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2261 % The trick with \xintApplyUnbraced is that for example when selecting
2262 % the elements smaller than pivot, if we had been using \xintApply we
2263 % would have had at the minimum an empty brace pair. Thus we use the
2264 % "unbraced" variant, but then the \QS@select@smaller has added in
2265 % anticipation a level of braces.
2266 \def\QS@sort@d #1#2{%
2267     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2268     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2269     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2270 }%
2271 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2272 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2273 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2274
2275 %
2276 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2277 %
2278 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2279 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2280 % latter must handle correctly an empty argument.
2281
2282 %-----
2283 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2284
2285 % This draws all with suitable highlighting for the newly chosen pivots
2286 % (which will be shown raised)

```

```

2287 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2288             \let\QSIr\DecoINERT
2289             \let\QSIrr\DecoINERT
2290             \let\QSRr\DecoRIGHTwithPivot
2291 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2292             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2293             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2294 }
2295
2296 % This sorts and then draws, showing where the pivot chosen in the previous
2297 % step go. Next time they will have become "inert". If pivotcount is still at
2298 % zero on exit from \QSpivotStep, then this is the signal to stop before
2299 % executing \QSSortStep.
2300 \def\QSSortStep {\def\QSLr {\noexpand\QS@sort@a}%
2301             \def\QSRr {\noexpand\QS@sort@a}%
2302             \def\QSIr {\noexpand\QSIrr}%
2303             \let\QSIrr\relax
2304             \edef\QS@list{\QS@list}%
2305             \let\QSLr\relax
2306             \let\QSRr\relax
2307             \let\QSIr\relax
2308             \edef\QS@list{\QS@list}%
2309             \let\QSLr\DecoLEFT
2310             \let\QSIr\DecoINERTwithPivot
2311             \let\QSIrr\DecoINERT
2312             \let\QSRr\DecoRIGHT
2313 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2314             \setcounter{cellcount}{0}%
2315             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2316 }
2317
2318 \def\QSinitialize #1{%
2319     % first, we convert the comma separated values into a list of braced items
2320     % we use an \edef, and anyhow many \edef's will be used later
2321     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2322     \let\QSRr\DecoRIGHT
2323     % The \QSRr marker mutated to draw the last element as
2324     % pivot and the earlier ones with the suitable style.
2325     %
2326     % The list of marked braced items \QS@list is used both for drawing
2327     % (as here) and for doing the exchange of elements during sort.
2328     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2329             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2330 }
2331

```

## 2.33 relationale-algebra.sty

2332 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2333 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]

2334 \RequirePackage{amsmath}

2335 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

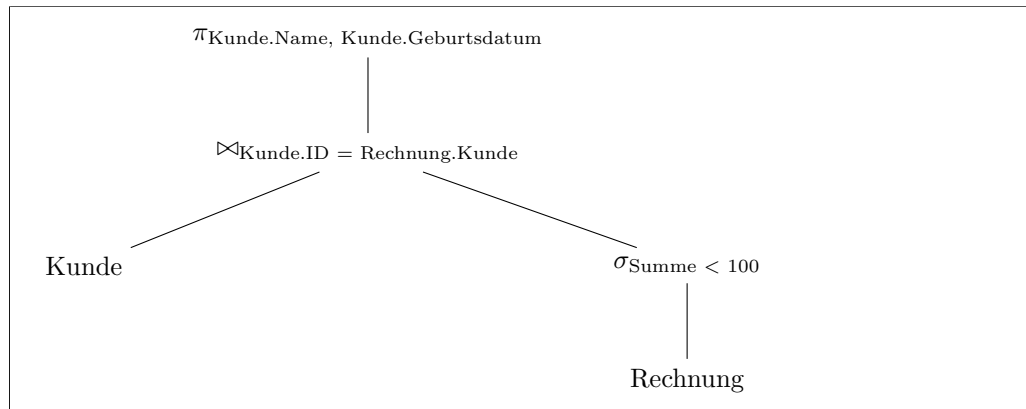
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

  \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}
```



2336 \RequirePackage{tikz}

2337 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

```
2338 \def\o@join{\setbox0=\hbox{\bowtie}}%
2339 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2340 }
```

\leftouterjoin A \leftouterjoin B:  $A \bowtie B$

2341 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B:  $A \bowtie B$

2342 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B:  $A \bowtie B$

2343 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}

2344

## 2.34 rmodell.sty

```

2345 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2346 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2347 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2348 Datenbanken.]
2349 \RequirePackage{soul}

```

### Let-Abkürzungen

```

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

```

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

```
2350 \def\liPrimaer#1{\ul{#1}}
```

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

```
2351 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```

2352 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2353 \ExplSyntaxOn
2354 \NewDocumentEnvironment { liRmodell }
2355 { +b }
2356 {
2357   \medskip
2358   {
2359     \linespread{2}
2360     \setlength{\parindent}{0pt}
2361     \li@Rmodell@Schrift#1
2362   }
2363   \medskip
2364 } {}
2365 \ExplSyntaxOff

```

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

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

```

2366 \def\liRelationMenge#1#2{
2367 \noindent
2368 #1 : \[ #2 ]\}
2369 \par
2370 }

```

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

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

```
2371 \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}

```

```
2372 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2373
```



## 2.35 sortieren.sty

```
2374 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2375 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2376 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}
}
```



```
2377 \RequirePackage{tikz}
2378 \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.

```
2379 \def\liVertauschen#1{
2380   \directlua{
2381     local sortieren = require('lehramt-informatik-sortieren')
2382     sortieren('#1')
2383   }
2384 }
```

`\liSortierPfeil`

```
2385 \def\liSortierPfeil#1#2{
2386   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2387 }
```

`\liSortierPfeilUnten`

```
2388 \def\liSortierPfeilUnten#1#2{
2389   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2390 }
```

`\liSortierMarkierung`

```
2391 \def\liSortierMarkierung#1#2{\node[
2392   draw,
2393   very thick,
2394   fit=(reihe.#1) (reihe.#2),
2395   inner sep=0pt
2396 ] {}
2397 }

2398 \tikzset{
2399   li sortierung zahlenreihe/.style={
2400     draw,
2401     thin,
2402     font=\large,
2403     rectangle split horizontal,
2404     rectangle split,
2405   }
2406 }
```

```

2407 % https://tex.stackexchange.com/a/140895
2408 \RequirePackage{forest,xstring}
2409 \usetikzlibrary{calc}
2410
2411 \makeatletter
2412 \pgfmathdeclarefunction{strrepeat}{2}{%
2413   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2414   \let\pgfmathresult\pgfutil@empty
2415   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2416     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2417     \advance\pgfmath@count-1\relax
2418   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2419 \makeatother
2420
2421 \def\myNodes{}
2422
2423 \ExplSyntaxOn
2424 \newcommand*\sortList[1]{%
2425   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2426 \ExplSyntaxOff
2427
2428 \forestset{
2429   sort/.code={%
2430     \pgfmathparse{level()>\forestSortLevel}%
2431     \ifnum\pgfmathresult=0
2432       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2433       \sortList\myList
2434       \StrSubstitute{\myList}{,}{ }{\myList}%
2435       \pgfmathparse{strrepeat("1",level())}%
2436       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-!({\forestov{name}}$)
2437         (m\forestov{name}) {\myList}}%
2438       \pgfmathparse{level()=\forestSortLevel}%
2439       \ifnum\pgfmathresult=1
2440         \forestOget{\forestov{@first}}{name}\forestFirst
2441         \forestOget{\forestov{@last}}{name}\forestLast
2442         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2443           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2444       \fi
2445       \ifnum\forestov{@parent}=0\else
2446         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2447       \fi
2448       \gappto\myNodes{;}%
2449     \fi}}
2450
2451 \forestset{sort level/.code=%
2452   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2453   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2454

```

## 2.36 spalten.sty

```
2455 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2456 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2457 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2458 realisiert werden kann.]
2459 \RequirePackage{multicol}
```

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

```
2460 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2461
```

## 2.37 struktogramm.sty

```
2462 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2463 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2464 Paket struktex zum Setzen von Struktogrammen]
2465 \RequirePackage{struktex}
2466
```

## 2.38 syntax.sty

```

2467 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2468 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2469 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2470 \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

```

```

2471 \ExplSyntaxOn
2472 \directlua{
2473   syntax = require('lehramt-informatik-syntax')
2474   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2475   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2476   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2477   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2478   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2479   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2480 }
2481 \RequirePackage{hyperref}
2482 \RequirePackage{minted}
2483 % pygmentize -L styles
2484 \usemintedstyle{colorful}
2485 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2486 %\AfterEndEnvironment{minted}{\end{mdframed}}
2487 %\setminted{breaklines=true,linenos}
2488 \setminted{
2489   breaklines=true,
2490   linenos,
2491   fontsize=\footnotesize,
2492 }

```

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

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

```

2493 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}

```

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

```

2494 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

2495 \def\li@GithubLink#1#2{
2496   \begin{flushright}
2497     \tiny
2498     Code-Beispiel~auf~Github~ansehen:~
2499     \href{#1}{\nolinkurl{#2}}
2500   \end{flushright}
2501 }

```

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

```

2502 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2503   \inputminted[#1]{java}{
2504     \directlua{
2505       syntax.drucke_absoluten_pfad('#2', false)
2506     }
2507   }
2508   \li@GithubLink

```

```

2509     {\directlua{syntax.drucke_github_url('#2', false)}}
2510     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2511 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2512 \NewDocumentCommand{\liJavaTestDatei}{ 0{firstline=3} m }{
2513   \inputminted[#1]{java}{
2514     \directlua{
2515       syntax.drucke_absoluten_pfad('#2', true)
2516     }
2517   }
2518   \li@GithubLink
2519   {\directlua{syntax.drucke_github_url('#2', true)}}
2520   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2521 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2522 \NewDocumentCommand{\liJavaExamen}{ 0{firstline=3} m m m m }{
2523   \inputminted[#1]{java}{
2524     \directlua{
2525       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2526     }
2527   }
2528
2529   \li@GithubLink
2530   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2531   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2532 }

\liAssemblerCode
2533 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2534 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2535   \inputminted{asm}{#1}
2536 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2537 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2538   \inputminted{componentpascal}{#1}
2539 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2540 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2541 \NewDocumentCommand{\liHaskellDatei}{ m }{
2542   \inputminted{haskell}{#1}
2543 }

2544 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2545 \def\liSqlCode#1{\mintinline{sql}|#1|}
2546

```

## 2.39 syntaxbaum.sty

```
2547 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2548 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2549 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2550 \RequirePackage{tikz-qtrees}
2551
2552 \tikzset{li parsetree/.style={
2553     every internal node/.style={
2554         draw,circle
2555     },
2556     every leaf node/.style={
2557         draw,rectangle
2558     },
2559 }
2560 }
2561
```

## 2.40 synthese-algorithmus.sty

```
2562 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2563 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2564 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2565 Relation in die 3. Normalform]

2566 \liLadePakete{normalformen,mathe,typographie}
2567 \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

```

2568 \def\liSyntheseUeberschrift#1{
2569   {
2570     \bfseries
2571     \sffamily
2572     \str_case:nn {#1} {
2573       {1} {Kanonische-Überdeckung}
2574       {1-1} {Linksreduktion}
2575       {1-2} {Rechtsreduktion}
2576       {1-3} {Löschen-leerer-Klauseln}
2577       {1-4} {Vereinigung}
2578       {2} {Relationsschemata-formen}
2579       {3} {Schlüssel-hinzufügen}
2580       {4} {Entfernung-überflüssiger-Teilschemata}
2581     }
2582   }
2583 }

```

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

```

2584 \def\li@synthese@erklaerung@texte#1{
2585   \str_case:nn {#1} {
2586     {1} {
2587       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2588       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2589       Schritten-erreicht-werden.
2590     }
2591     {1-1} {
2592       Führe-für-jede-funktionale-Anhängigkeit-
2593       $\alpha\rightarrow\beta$-in-F$-die-Linksreduktion-durch,-
2594       überprüfe-also-für-alle-
2595       $A$-in-$\alpha$,~ob-$A$-überflüssig-ist,~d.h.-ob-
2596       $\beta\subseteq A$-
2597     }
2598     {1-2} {
2599       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2600       alle-$B$-in-$\beta$,~ob-$B$-in-$\alpha$-
2601       $\alpha\rightarrow\beta$-gilt.-In-diesem-Fall-ist-$B$-auf-der-rechten-Seite-
2602       überflüssig-und-kann-eliminiert-werden,~dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow\beta$-
2603       ersetzt.
2604     }
2605     {1-3} {
2606       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2607       entstanden-sind.
2608     }
2609     {1-4} {
2610       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2611       der-Form-$\alpha\rightarrow\beta_1,\dots,\alpha\rightarrow\beta_n$,~so-dass-$\alpha\rightarrow\beta_1\cup\dots\cup\beta_n$-
2612       verbleibt.
2613     }
2614     {2} {
2615       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-in-F$\subseteq R$-ein-Relationenschema-$\mathcal{R}_\alpha$-
2616     }
2617   }
2618 }

```

```

2623      :=~\alpha~\cup~\beta$.
2624    }
2625    {3} {
2626      Falls-eines-der-in-Schritt-2.-erzeugten-Schemata- $R_{\alpha}$ -
2627      einen-Schlüsselkandidaten-von- $R$ -bezüglich- $F_c$ -
2628      enthält,~sind-wir-fertig,~sonst-wähle-einen-Schlüsselkandidaten-
2629       $K \subseteq R$ -aus-und-definiere-folgendes-
2630      zusätzliche-Schema:~ $R_{\alpha} \setminus K := \emptyset$ -
2631      und- $F_{\alpha} \setminus K := \emptyset$ 
2632    }
2633    {4} {
2634      Eliminiere-diejenigen-Schemata- $R_{\alpha}$ ,~die-in-einem-
2635      anderen-Relationenschema- $R_{\alpha'}$ -enthalten-sind,~d.h.~
2636       $R_{\alpha} \subseteq R_{\alpha'}$ .
2637    }
2638  }
2639 }
2640 \def\liSyntheseErklaerung#1{
2641   {
2642     \itshape
2643     \footnotesize
2644     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2645   }
2646 }

```

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

```

2647 \def\liSyntheseUeberErklaerung#1{
2648   \liSyntheseUeberschrift{#1}\par
2649   \liSyntheseErklaerung{#1}
2650 }

2651 \ExplSyntaxOff
2652

```

## 2.41 tabelle.sty

```
2653 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2654 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
2655 \RequirePackage{tabularx}
2656
```

## 2.42 typographie.sty

```
2657 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2658 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2659 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2660 formatierung.sty definiert.]
```

```
2661 \ExplSyntaxOn
```

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

```
2662 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2663 \let\liErledigt=\faCheckSquareO
```

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

```
2664 \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 —

```
2665 \def\liParagraphMitLinien#1{
```

```
2666 \noindent
```

```
2667 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2668 \enspace
```

```
2669 #1
```

```
2670 \enspace
```

```
2671 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2672 \par
```

```
2673 \medskip
```

```
2674 }
```

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

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

```
2675 \def\liGeschweifteKlammern#1#2#3#4{
```

```
2676 \par
```

```
2677 \medskip
```

```
2678 \noindent
```

```
2679 #1 \, \$= \Bigl\{ \$
```

```
2680 \vspace{#3}
```

```
2681 #2
```

```
2682 \vspace{#4}
```

```
2683 \begin{flushright} \$\Bigr\} \$\end{flushright}
```

```
2684 \par
```

```
2685 }
```

```
2686 \ExplSyntaxOff
```

```
2687
```

## 2.43 uml.sty

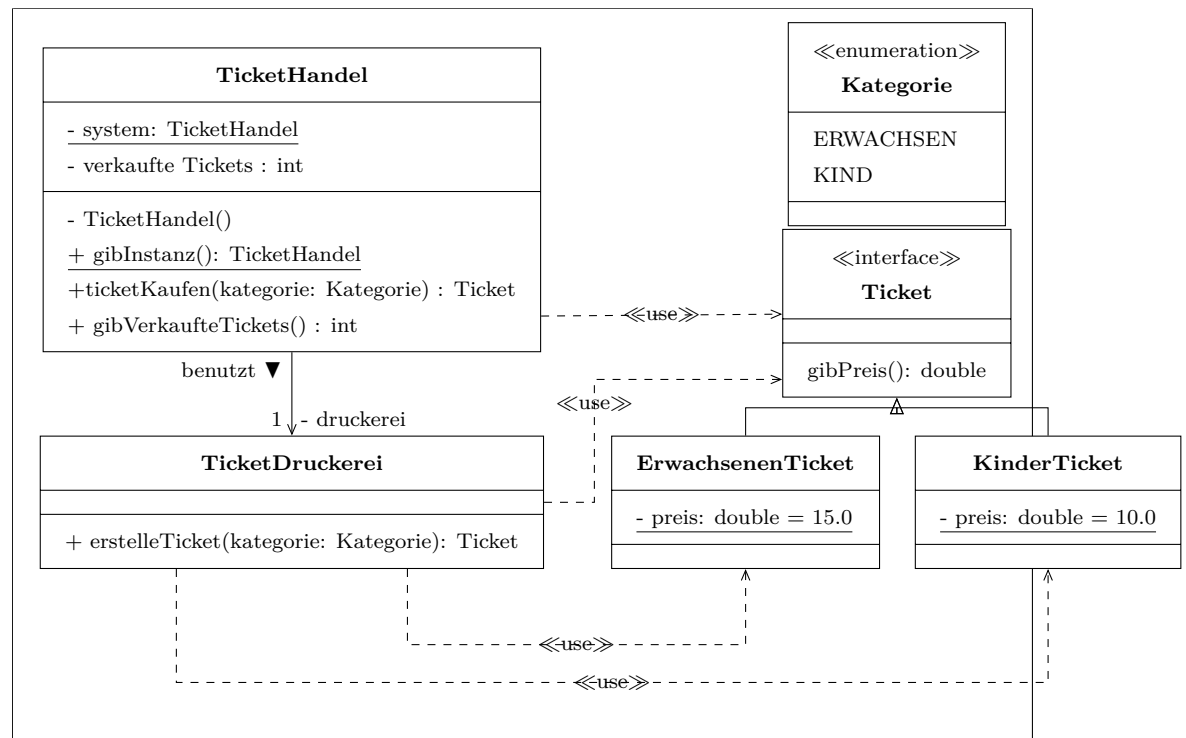
```

2688 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2689 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2690 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2691 Erweiterung bereitstellt]

2692 \RequirePackage{tikz-uml}
2693 \RequirePackage{tikz-uml-activity}
2694 % Not compatible with wasysym
2695 %\RequirePackage{mathabx}
2696 \RequirePackage{wasysym}
2697 \usetikzlibrary{positioning}

2698 \tikzumlset{
2699   fill class=white!0,
2700   font=\footnotesize,
2701   fill object=white!0,
2702   fill note=white!0,
2703   fill state=white!0,
2704   % Use case
2705   fill usecase=white!0,
2706   fill system=white!0,
2707 }

```



```

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

```

```

2708 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2709   \def\@liDirLeft{}
2710   \def\@liDirRight{ \RIGHTarrow}
2711   \pgfkeys{/lese/dir/.is choice}
2712   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2713   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2714   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2715   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2716
2717   \def\@liPos{above}
2718   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2719

```

```

2720 \def\@liDistance{0cm}
2721 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2722
2723 \pgfkeys{/lese/.cd,#1}
2724
2725 \node[\@liPos = \@liDistance of #2-middle] {
2726   \@liDirLeft{\footnotesize#2}\@liDirRight
2727 };
2728 }
2729

```

## 2.44 vollstaendige-induktion.sty

2730 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2731 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01  
 2732 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die  
 2733 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{\e{Java nach Mathe}}\\
\%
&= \frac{
  \{(4\m{n} + 2) \cdot \text{cn}(\m{n})\}
  \{\m{n} + 2\}
&\text{\e{addiert, subtrahiert}}\\
\%
&= \frac{
  \{(4n + 2) \cdot \m{(2n)!}\}
  \{(n + 2) \cdot \m{(n + 1)!} \cdot n!\}
&\text{\e{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{\e{\$(n + 1)\$ multipliziert}} \\
\%
&= \frac{
  \{(4n + 2) \cdot \m{(n + 1) \cdot (2n)!}\}
  \{(n + 2) \cdot (n + 1)! \cdot \m{(n + 1) \cdot n!}\}
&\text{\e{umsortiert}} \\
\%
&= \frac{
  \{\m{(2(n + 1))!}\}
  \{\m{(n + 2)! \cdot (n + 1)!}\}
&\text{\e{Hilfsgleichungen verwendet}}\\
\%
&= \frac{
  \{(2(\m{n} + 1))!\}
  \{((\m{n} + 1) + 1)! \cdot (\m{n} + 1)!\}
&\text{\e{\$(n + 1)\$ verdeutlicht}}\\
\end{align*}
```

Lade häufig benötigte Pakete

2734 \RequirePackage{lehramt-informatik-typographie}  
 2735 \RequirePackage{lehramt-informatik-mathe}  
 2736 \RequirePackage{lehramt-informatik-syntax}  
 2737 \ExplSyntaxOn

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

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

2738 \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

```
2739 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2740 \def\liInduktionAnfang{
2741   \liPseudoUeberschrift{Induktionsanfang}
2742
2743   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2744   \liParagraphMitLinien{
2745     Beweise,~dass~$A(1)$~eine~wahre~Aussage~ist.
2746   }
2747 }
```

\liInduktionVoraussetzung

```
2748 \def\liInduktionVoraussetzung{
2749   \liPseudoUeberschrift{Induktionsvoraussetzung}
2750
2751   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2752   \liParagraphMitLinien{
2753     Die~Aussage~$A(k)$~ist~wahr~für~ein~beliebiges~$k$ \in \mathbb{N}$$.
2754   }
2755 }
```

\liInduktionSchritt

```
2756 \def\liInduktionSchritt{
2757   \liPseudoUeberschrift{Induktionsschritt}
2758
2759   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2760   \liParagraphMitLinien{
2761     Beweise,~dass~wenn~$A(n=k)$~wahr~ist,~
2762     auch~$A(n=k+1)$~wahr~sein~muss.
2763   }
2764 }
```

```
2765 \ExplSyntaxOff
```

```
2766
```



## 2.45 wasserfall.sty

```
2767 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2768 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2769 \RequirePackage{tikz}
2770 \tikzset{wasserfall/.style={
2771   >=stealth,
2772   node distance = 2mm and -8mm,
2773   start chain = A going below right,
2774   every node/.style = {
2775     draw,
2776     text width=24mm,
2777     minimum height=12mm,
2778     align=center,
2779     inner sep=1mm,
2780     fill=white,
2781     drop shadow={fill=black},
2782     on chain=A
2783   },
2784 }}
2785 \usetikzlibrary{chains,positioning,shadows}
2786
```

## 2.46 wpkalkuel.sty

2787 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2788 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

### Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2789 \RequirePackage{amsmath}

2790 \ExplSyntaxOn

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

2791 \def\liWpKalkuelOhneMathe#1#2{

2792 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2793 }

2794 \def\liWpKalkuel#1#2{

2795 \ifmmode

2796 \liWpKalkuelOhneMathe{#1}{#2}

2797 \else

2798 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2799 \fi

2800 }

\MatheEnv

2801 \def\MatheEnv#1{

2802 \medskip

2803

2804 \hspace{1em}#1

2805

2806 \medskip

2807 }

\Mathe

2808 \def\Mathe#1{

2809 \MatheEnv{#1\$}

2810 }

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

2811 \def\liWpEquivalent#1{

2812 \MatheEnv{\$\equiv\$\hspace{1em}\$#1\$}

2813 }

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

2814 \newlength{\@Skip@Erklaerung@Reset}

2815 \def\liWpErklaerung#1{

2816 \setlength{\@Skip@Erklaerung@Reset}{\leftskip}

2817 \setlength{\leftskip}{0.5cm}

2818

2819 \par

2820 \noindent

2821 {

2822 \scriptsize

2823 #1

2824 }

2825 \par

2826

2827 \setlength{\leftskip}{\@Skip@Erklaerung@Reset}

2828 }

\liWpErklaerungVerzweigung

```

2829 \def\liWpErklaerungVerzweigung{
2830   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2831   \equiv
2832   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2833   \lor
2834   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2835 }

2836 \ExplSyntaxOff

2837

```

### 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.

Symbols			
\#	111	\advance	2417
\,	345, 400, 1184, 1186, 1687, 1688, 1689, 2023, 2493, 2679	\AfterEndEnvironment	2486
\@Skip@Erklaerung@Reset	2814, 2816, 2827	\allsectionsfont	1233
\@afterheading	1654	\Alph	1247
\@afterindentfalse	1654	\alph	1247, 1248
\@liDirLeft	2709, 2714, 2726	\alpha	2593, 2595, 2596, 2599, 2601, 2602, 2603, 2604, 2605, 2609, 2615, 2616, 2621, 2622, 2623, 2626, 2634, 2635, 2636
\@liDirRight	2710, 2712, 2713, 2714, 2715, 2726	\arabic	1247, 2212, 2217, 2222, 2228, 2234, 2240
\@liDistance	2720, 2721, 2725	\arraystretch	1913
\@liPos	2717, 2718, 2725	<b>C</b>	
\\	610, 633, 634, 637, 638, 641, 642, 734, 735, 736, 843, 872, 874, 900, 909, 954, 996, 997, 998, 1003, 1004, 1005, 1025, 1561, 1918, 1976, 1979	\c	1330, 1331
\{	209, 1121, 1131, 1143, 1144, 1149, 1183, 1402, 1963, 2368, 2679, 2830	\cdot	1731, 1784, 1795
\}	209, 1121, 1131, 1143, 1144, 1151, 1187, 1403, 1963, 2368, 2683, 2830	\centerline	1361, 2291, 2313, 2328
\_	27, 35, 44, 46, 48, 295, 320, 344, 345, 359, 360, 366, 369, 372, 382, 394	\chapter	1237, 1238
\_	1975, 1985	\char	1561
<b>A</b>		\clearpage	1555
\addbibresource	1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538, 1539, 1540	\cline	610
		\clist	228, 272, 273, 292, 296, 2425
		\columnbreak	2460
		\cs	295, 320, 344, 345, 382, 394, 1637
		\csname	1315, 1318
		\cup	1144, 1997, 2602, 2616, 2623
		<b>D</b>	
		\DeclareMathSymbol	1861, 1862
		\DecoINERT	2215, 2288, 2289, 2311
		\DecoINERTwithPivot	2231, 2310
		\DecoLEFT	2210, 2309
		\DecoLEFTwithPivot	2225, 2287
		\DecoRIGHT	2220, 2312, 2322
		\DecoRIGHTwithPivot	2237, 2290
		\beta	2593, 2596, 2600, 2601, 2602, 2605, 2615, 2616, 2617, 2622, 2623
		\bf	2190, 2191, 2192
		\BeforeBeginEnvironment	2485
		\begin	631, 682, 697, 732, 756, 803, 835, 850, 870, 880, 896, 916, 948, 965, 994, 1019, 1040, 1055, 1164, 1253, 1328, 1356, 1365, 1372, 1480, 1553, 1605, 1610, 1618, 1643, 1648, 1659, 1669, 1673, 1756, 1760, 1776, 1799, 1822, 1837, 1916, 1917, 1967, 2012, 2132, 2152, 2293, 2315, 2329, 2485, 2496, 2683
		\begingroup	1554, 1965, 2413
		\beschriftung	1585, 1589, 1593, 1597, 1601, 1603





\liEntwurfsAbstrakteFabrik	\liEntwurfsModellPraesentation	\liSteuerDiagramm	104
..... 677	..... 1018, 1036	\liKellerKante	142
\liEntwurfsAbstrakteFabrikCode	\liEntwurfsZustand	\liKellerUebergang	..
..... 672, 679	\liEntwurfsZustandAkteure	..... 137, 143	
\liEntwurfsAbstrakteFabrikUml	..... 1054, 1074	\liKontrollCode	1491
..... 630, 678	\liEntwurfsZustandUml	liKontrollflussgraph	
\liEntwurfsAdapter	..... 1039, 1073	(environment)	1479
\liEntwurfsAdapterAkteure	\liEpsilon	\liKontrollKnotenPfad	..... 1493
..... 696, 728	\liErAttribute	..... 1492, 1497	
\liEntwurfsAdapterCode	... 1087, 1101, 1103	\liKurzeTabellenLinie	610
..... 720, 729	\liErDatenbankName	\liLadeAllePakete	.. 230
\liEntwurfsAdapterUml	\liErEntity	\liLadePakete	56,
..... 681, 727	\liErledigt	59, 226, 231, 486,	
\liEntwurfsBeobachter	\liErMpAttribute	547, 1120, 1351,	
\liEntwurfsBeobachterAkteure	\liErMpEntity	1867, 1947, 2095, 2566	
..... 755, 799	\liErMpRelationship	\liLatexCode	2494
\liEntwurfsBeobachterCode	\liErRelationship	\liLeereZelle	1905
..... 789, 800	... 1086, 1095, 1097	liLernkartei (environ-	
\liEntwurfsBeobachterUml	\liExamensAufgabe	ment) .....	1657
..... 731, 798	\liExamensAufgabeA	\liLinksReduktion	.. 1972
\liEntwurfsDekorierer	\liExamensAufgabeTA	\liLinksReduktionInline	..... 1981, 1989
\liEntwurfsDekoriererAkteure	\liExamensAufgabeTTA	\liMasterExkurs	1836
..... 831	liExkurs (environment)	\liMasterFaelle	1775, 1843
\liEntwurfsDekoriererCode	\liFalsch	\liMasterFallRechnung	..... 1821
..... 821, 832	\liFlaci	liMasterVariablen	..
\liEntwurfsDekoriererUml	\liFremd	..... 1755, 1838	
..... 802, 830	\liFunktionaleAbhaengigkeit	\liMasterVariablenDeklaration	..... 1798
\liEntwurfsEinfacheFabrik	... 1994, 1997, 2005	\liMasterWolframLink	1846
..... 865	\liFunktionaleAbhaengigkeiten	\liMenge	73, 74, 76,
\liEntwurfsEinfacheFabrikAkteure	..... 2008	115, 116, 117, 121,	
..... 849, 867	\liFussnote	173, 174, 175, 179,	
\liEntwurfsEinfacheFabrikUml	... 1868, 1870	1121, 1170, 1209, 1210	
..... 834, 866	\liFussnoteDreiText	\liMengeOhneMathe	...
\liEntwurfsEinzelstueck	..... 1882, 1900	1121, 1124, 1126	
..... 890	\liFussnoteEinsText	\liMinimierungErklaerung	..... 1927
\liEntwurfsEinzelstueckAkteure	..... 1874, 1894	\liMinispracheDatei	2537
..... 879, 892	\liFussnoteLink	\linespread	..... 2359
\liEntwurfsEinzelstueckCode	..... 1684	\liNichtsZuTun	..... 2664
..... 887, 893	liFussnoten	\liO	..... 1718, 1746
\liEntwurfsEinzelstueckUml	\liFussnoteUrl	\liOmega	..... 1708, 1752
..... 869, 891	... 1033, 1680	\liOmegaOhneMathe	...
\liEntwurfsErbauer	\liFussnoteVierText	... 1708, 1713, 1715	
\liEntwurfsErbauerAkteure	..... 1886, 1903	\liOOhneMathe	.....
..... 915, 945	\liFussnoteZweiText	... 1718, 1723, 1725	
\liEntwurfsErbauerUml	..... 1878, 1897	\liParagraphMitLinien	...
..... 895, 944	\liGeschweifteKlammern	535, 1929, 2644,	
\liEntwurfsFabrikmethode	... 1162, 2009, 2675	2665, 2744, 2752, 2760	
..... 989, 1014	\liGrammatik	\liPetriErreichKnotenDrei	..... 2089
\liEntwurfsFabrikmethodeAkteure	liGraphenFormat (envi-	\liPetriErreichTransition	..... 2086
..... 964, 991	ronment) .... 1307	\liPetriSetzeSchluessel	..... 2031
\liEntwurfsFabrikmethodeUml	\liHanoi	\liPetriTransitionsName	..... 2078, 2090
..... 947, 990	\liHaskellCode	\liPetriTransitionsNameOhneMathe	... 2078, 2081, 2083
\liEntwurfsKompositum	\liHaskellDatei		
..... 1014	... 2541		
\liEntwurfsKompositumAkteure	\liInduktionAnfang		
..... 1016	\liInduktionErklaerung		
\liEntwurfsKompositumUml	..... 2739		
..... 993, 1015	\liInduktionMarkierung		
\liEntwurfsModellPraesentation	..... 2738		
..... 1035	\liInduktionSchritt		
\liEntwurfsModellPraesentation	2756		
..... 1037	\liInduktionVoraussetzung		
	..... 2748		
	\liJavaCode		
	..... 2493		
	\liJavaExamen		
	..... 2522		
	\liJavaTestAkteure		
	..... 2512		
	liKasten (environment)		
	1252		

<code>\liPetriTransPfeile</code> <a href="#">2090</a>	<code>\liTheta</code> ..... <a href="#">1698</a> ,	<code>\loop</code> ..... <a href="#">2172</a>	
<code>\liPolynomiellReduzierbar</code> ..... <a href="#">1371</a>	<a href="#">1749</a> , <a href="#">1778</a> , <a href="#">1784</a> , <a href="#">1789</a>	<code>\lor</code> ..... <a href="#">2833</a>	
<code>\liPotenzmenge</code> ..... ... <a href="#">1130</a> , <a href="#">1134</a> , <a href="#">2101</a>	<code>\liThetaOhneMathe</code> ... ... <a href="#">1698</a> , <a href="#">1703</a> , <a href="#">1705</a>	<code>\ltimes</code> ..... <a href="#">1854</a>	
<code>\liPotenzmengeOhneMathe</code> ... <a href="#">1131</a> , <a href="#">1132</a> , <a href="#">1133</a>	<code>\liTOhneMathe</code> ..... ... <a href="#">1728</a> , <a href="#">1737</a> , <a href="#">1739</a>	M	
<code>\liPrimaer</code> ..... <a href="#">2350</a>	<code>\liTuringKante</code> ..... <a href="#">205</a>	<code>\makeatletter</code> .. <a href="#">1653</a> , <a href="#">2411</a>	
<code>\liProblemBeschreibung</code> ..... <a href="#">1355</a>	<code>\liTuringLeerzeichen</code> ..... <a href="#">161</a> , <a href="#">169</a>	<code>\makeatother</code> .. <a href="#">1655</a> , <a href="#">2419</a>	
<code>\liProblemClique</code> ... <a href="#">1378</a>	<code>\liTuringMaschine</code> ... <a href="#">162</a>	<code>\marginpar</code> ..... <a href="#">1090</a> , <a href="#">1096</a> , <a href="#">1102</a> , <a href="#">1560</a>	
<code>\liProblemName</code> ..... ..... <a href="#">1354</a> , <a href="#">1361</a> , <a href="#">1373</a> , <a href="#">1375</a> , <a href="#">1388</a> , <a href="#">1399</a> , <a href="#">1400</a> , <a href="#">1408</a> , <a href="#">1409</a>	<code>\liTuringUeberfuehrung</code> ..... <a href="#">208</a>	<code>\mathbb</code> .. <a href="#">1409</a> , <a href="#">2159</a> , <a href="#">2753</a>	
<code>\liProblemSat</code> ..... <a href="#">1407</a>	<code>\liTuringUebergaenge</code> ..... <a href="#">200</a> , <a href="#">206</a>	<code>\mathbin</code> . <a href="#">2341</a> , <a href="#">2342</a> , <a href="#">2343</a>	
<code>\liProblemSubsetSum</code> . ..... <a href="#">1398</a> , <a href="#">1407</a>	<code>\liTuringUebergangZelle</code> ..... <a href="#">195</a>	<code>\mathcal</code> .. <a href="#">1719</a> , <a href="#">2622</a> , <a href="#">2627</a> , <a href="#">2629</a> , <a href="#">2630</a> , <a href="#">2631</a>	
<code>\liProblemVertexCover</code> ..... <a href="#">1378</a> , <a href="#">1386</a>	<code>\liUeberfuehrungsFunktion</code> ..... <a href="#">1135</a>	<code>\Mathe</code> ..... <a href="#">2808</a>	
<code>\liProduktionen</code> <a href="#">1169</a> , <a href="#">1211</a>	<code>\liUeberfuehrungsFunktionOhneMathe</code> ... <a href="#">1135</a> , <a href="#">1138</a> , <a href="#">1140</a>	<code>\MatheEnv</code> <a href="#">2801</a> , <a href="#">2809</a> , <a href="#">2812</a>	
<code>liProduktionsRegeln</code> (environment) <a href="#">1159</a>	<code>liUebergangsTabelle</code> (environment) <a href="#">1913</a>	<code>\mathord</code> ..... <a href="#">1861</a> , <a href="#">1862</a>	
<code>liProjektSprache</code> (envi- ronment) .... <a href="#">1577</a>	<code>\liUeberschriftDreiecksTabelle</code> ..... <a href="#">1923</a>	<code>\mdfsetup</code> ..... <a href="#">1251</a> , <a href="#">1586</a> , <a href="#">1590</a> , <a href="#">1594</a> , <a href="#">1598</a>	
<code>\liPseudoUeberschrift</code> ..... <a href="#">1562</a> , <a href="#">1612</a> , <a href="#">1613</a> , <a href="#">1915</a> , <a href="#">1925</a> , <a href="#">2741</a> , <a href="#">2749</a> , <a href="#">2757</a>	<code>\liUmlLeserichtung</code> . <a href="#">2708</a>	<code>\medskip</code> ..... <a href="#">1363</a> , <a href="#">1571</a> , <a href="#">1573</a> , <a href="#">1647</a> , <a href="#">1675</a> , <a href="#">2357</a> , <a href="#">2363</a> , <a href="#">2673</a> , <a href="#">2677</a> , <a href="#">2802</a> , <a href="#">2806</a>	
<code>\liPumpingKontextfrei</code> ..... <a href="#">2147</a>	<code>\liVertauschen</code> ..... <a href="#">2379</a>	<code>\emph</code> ..... <a href="#">1560</a>	
<code>\liPumpingRegulaer</code> . <a href="#">2126</a>	<code>\liWortInSprache</code> .... <a href="#">611</a>	<code>\mintinline</code> .... <a href="#">2493</a> , <a href="#">2494</a> , <a href="#">2533</a> , <a href="#">2540</a> , <a href="#">2545</a>	
<code>liQuellen</code> (environment) ..... <a href="#">1637</a>	<code>\liWortNichtInSprache</code> <a href="#">616</a>	<code>\mkern</code> ... <a href="#">2341</a> , <a href="#">2342</a> , <a href="#">2343</a>	
<code>\liRechtsReduktionInline</code> ..... <a href="#">1989</a>	<code>\liWpEquivalent</code> .... <a href="#">2811</a>	<code>\mlq</code> ..... <a href="#">1859</a> , <a href="#">1861</a>	
<code>\liRekursionsGleichung</code> ..... <a href="#">1742</a> , <a href="#">1802</a>	<code>\liWpErklaerung</code> .... <a href="#">2814</a>	<code>\mrq</code> ..... <a href="#">1859</a> , <a href="#">1862</a>	
<code>\liRelation</code> ..... <a href="#">2019</a>	<code>\liWpErklaerungVerzweigung</code> ..... <a href="#">2829</a>	<code>\msg</code> ..... <a href="#">39</a> , <a href="#">405</a>	
<code>liRelationenSchemaFormat</code> (environment) <a href="#">2372</a>	<code>\liWpKalkuel</code> ..... <a href="#">2791</a>	<code>\myList</code> ..... <a href="#">2432</a> , <a href="#">2433</a> , <a href="#">2434</a> , <a href="#">2437</a>	
<code>\liRelationMenge</code> ... <a href="#">2366</a>	<code>\liWpKalkuelOhneMathe</code> ..... <a href="#">2791</a> , <a href="#">2796</a> , <a href="#">2798</a> , <a href="#">2830</a> , <a href="#">2832</a> , <a href="#">2834</a>	<code>\myNodes</code> ..... <a href="#">2421</a> , <a href="#">2436</a> , <a href="#">2442</a> , <a href="#">2446</a> , <a href="#">2448</a>	
<code>\liRichtig</code> ..... <a href="#">477</a>	<code>\liZustandsBuchstabe</code> ..... <a href="#">1145</a> , <a href="#">1154</a> , <a href="#">1156</a> , <a href="#">1174</a> , <a href="#">1176</a>	N	
<code>liRmodell</code> (environment) ..... <a href="#">2352</a>	<code>\liZustandsBuchstabeGross</code> ... <a href="#">1146</a> , <a href="#">1155</a> , <a href="#">1157</a>	<code>\NeedsTeXFormat</code> ..... <a href="#">1</a> , <a href="#">19</a> , <a href="#">54</a> , <a href="#">222</a> , <a href="#">412</a> , <a href="#">473</a> , <a href="#">480</a> , <a href="#">544</a> , <a href="#">607</a> , <a href="#">622</a> , <a href="#">1077</a> , <a href="#">1113</a> , <a href="#">1226</a> , <a href="#">1266</a> , <a href="#">1275</a> , <a href="#">1280</a> , <a href="#">1309</a> , <a href="#">1347</a> , <a href="#">1419</a> , <a href="#">1502</a> , <a href="#">1517</a> , <a href="#">1523</a> , <a href="#">1545</a> , <a href="#">1691</a> , <a href="#">1851</a> , <a href="#">1864</a> , <a href="#">1943</a> , <a href="#">2026</a> , <a href="#">2092</a> , <a href="#">2117</a> , <a href="#">2122</a> , <a href="#">2166</a> , <a href="#">2332</a> , <a href="#">2345</a> , <a href="#">2374</a> , <a href="#">2455</a> , <a href="#">2462</a> , <a href="#">2467</a> , <a href="#">2547</a> , <a href="#">2562</a> , <a href="#">2653</a> , <a href="#">2657</a> , <a href="#">2688</a> , <a href="#">2730</a> , <a href="#">2767</a> , <a href="#">2787</a>	
<code>\liRundeKlammer</code> . <a href="#">1695</a> , <a href="#">1699</a> , <a href="#">1709</a> , <a href="#">1719</a> , <a href="#">1733</a>	<code>\liZustandsmenge</code> ... <a href="#">1133</a>	<code>\neg</code> ..... <a href="#">2834</a>	
<code>\liSortierMarkierung</code> <a href="#">2391</a>	<code>\liZustandsmengeNr</code> .. ..... <a href="#">1147</a> , <a href="#">2110</a>	<code>\negthinspace</code> ..... <a href="#">1696</a>	
<code>\liSortierPfeil</code> .... <a href="#">2385</a>	<code>\liZustandsmengeNrGross</code> ..... <a href="#">1155</a>	<code>\newcounter</code> ... <a href="#">2186</a> , <a href="#">2187</a>	
<code>\liSortierPfeilUnten</code> <a href="#">2388</a>	<code>\liZustandsMengenSammlung</code> ..... <a href="#">2097</a>	<code>\NewDocumentCommand</code> . <a href="#">64</a> , <a href="#">105</a> , <a href="#">142</a> , <a href="#">163</a> , <a href="#">205</a> , <a href="#">226</a> , <a href="#">549</a> , <a href="#">566</a> , <a href="#">611</a> , <a href="#">616</a> , <a href="#">1180</a> , <a href="#">1202</a> , <a href="#">1371</a> , <a href="#">1494</a> , <a href="#">1521</a> , <a href="#">1680</a> , <a href="#">1684</a> , <a href="#">2008</a> , <a href="#">2019</a> , <a href="#">2086</a> , <a href="#">2502</a> , <a href="#">2512</a> , <a href="#">2522</a> , <a href="#">2534</a> , <a href="#">2537</a> , <a href="#">2541</a> , <a href="#">2708</a>	
<code>\liSpaltenUmbruch</code> .. <a href="#">2460</a>	<code>\liZustandsMengenSammlungNr</code> ..... <a href="#">2106</a>		
<code>\liSqlCode</code> ..... <a href="#">2545</a>	<code>\liZustandsmengeOhneMathe</code> ..... <a href="#">1133</a>		
<code>\listen@punkt</code> .. <a href="#">1637</a> , <a href="#">1649</a>	<code>\liZustandsname</code> .... <a href="#">1156</a>		
<code>\liStrich</code> ..... <a href="#">1353</a>	<code>\liZustandsnameGross</code> ... <a href="#">1157</a> , <a href="#">2098</a> , <a href="#">2107</a>		
<code>\liSyntheseErklaerung</code> ..... <a href="#">2584</a> , <a href="#">2649</a>	<code>\liZustandsnameTiefgestellt</code> ..... <a href="#">1172</a>		
<code>\liSyntheseUeberErklaerung</code> ..... <a href="#">2647</a>	<code>\liZustandsPaar</code> .... <a href="#">1907</a>		
<code>\liSyntheseUeberschrift</code> ..... <a href="#">2568</a> , <a href="#">2648</a>	<code>\liZustandsPaarVariablenName</code> ... <a href="#">1906</a> , <a href="#">1909</a> , <a href="#">1910</a>		
<code>\liT</code> <a href="#">1728</a> , <a href="#">1743</a> , <a href="#">1757</a> , <a href="#">1818</a>	<code>\llap</code> ..... <a href="#">2339</a>		
<code>\liTeilen</code> ..... <a href="#">1952</a>	<code>\log</code> ..... <a href="#">1746</a> , <a href="#">1749</a> , <a href="#">1752</a> , <a href="#">1778</a> , <a href="#">1784</a>	<code>\NewDocumentEnvironment</code> <a href="#">1159</a> , <a href="#">1252</a> , <a href="#">1307</a> ,	
<code>\literatur</code> .... <a href="#">1519</a> , <a href="#">1543</a>			



1479, 1577, 1578, 1581, 1608, 1616, 1638, 1657, 1667, 1914, 1964, 2354, 2372	\ProvidesPackage . . . . 2, 20, 55, 223, 413, 474, 481, 545, 608, 623, 1078, 1114, 1227, 1267, 1276, 1281, 1310, 1348, 1420, 1503, 1518, 1524, 1546, 1692, 1852, 1865, 1944, 2027, 2093, 2118, 2123, 2167, 2333, 2346, 2375, 2456, 2463, 2468, 2548, 2563, 2654, 2658, 2689, 2731, 2768, 2788	476, 546, 625, 626, 1080, 1082, 1083, 1119, 1228, 1229, 1232, 1234, 1236, 1241, 1250, 1257, 1268, 1269, 1278, 1282, 1283, 1284, 1312, 1313, 1352, 1421, 1520, 1525, 1526, 1542, 1549, 1550, 1551, 1579, 1694, 1855, 1856, 2028, 2030, 2120, 2180, 2182, 2334, 2335, 2336, 2349, 2377, 2408, 2459, 2465, 2470, 2481, 2482, 2550, 2655, 2662, 2692, 2693, 2695, 2696, 2734, 2735, 2736, 2769, 2789
\newlength . . . . . 2814		
\node . . . . . 562, 1487, 1492, 2212, 2217, 2222, 2228, 2234, 2240, 2391, 2436, 2725		
\noexpand . . . . . 2300, 2301, 2302, 2321, 2436		
\noindent . . . . . 357, 613, 618, 1564, 1566, 1570, 1574, 1602, 1630, 1632, 1645, 1661, 1663, 1671, 1840, 1893, 1896, 1899, 1902, 2367, 2666, 2678, 2820		
\nolinkurl . . . . . 2499		
\normalsize . . . . . 1239		
\notin . . . . . 619		
\null . . . . . 2671		
<b>O</b>		
\o@join . . . . . 2338, 2341, 2342, 2343		
\Omega . . . . . 1709		
\omega 2128, 2129, 2149, 2150		
\or . . . . . 2249		
<b>P</b>		
\pagestyle . . . . . 1262		
\par . . . 356, 377, 539, 1191, 1199, 1569, 1631, 1654, 1662, 2291, 2313, 2328, 2369, 2648, 2672, 2676, 2684, 2819, 2825		
\paragraph . . . . . 1239		
\parindent . . . . . 2360		
\path . . . 96, 143, 206, 577		
\pgfkeys . . 2042, 2711, 2712, 2713, 2714, 2715, 2718, 2721, 2723		
\pgfmath@count . . . . . . . . 2413, 2415, 2417		
\pgfmath@smuggleone 2418		
\pgfmathdeclarefunction . . . . . 2412		
\pgfmathhint . . . . . 2413		
\pgfmathparse . . . . . . . . . 1321, 2430, 2435, 2438, 2452, 2453		
\pgfmathresult . . . . . . . . . 1322, 2413, 2414, 2416, 2418, 2431, 2439, 2452, 2453		
\pgfutil@empty . . . . 2414		
\pgfutil@loop . . . . 2415		
\pgfutil@repeat . . . 2418		
\preceq . . . . . 1374		
\prime . . . . . 1353		
\printbibliography . 1543		
	<b>Q</b>	
	\QS@list . . . . . 2293, 2304, 2308, 2315, 2321, 2326, 2329	
	\QS@select@equal . . . . . . . . . 2268, 2272	
	\QS@select@greater . . . . . . . 2269, 2273	
	\QS@select@smaller . . . . . 2264, 2267, 2271	
	\QS@sort@a . . . . . 2246, 2279, 2300, 2301	
	\QS@sort@b . . . 2246, 2247	
	\QS@sort@c . . . 2250, 2257	
	\QS@sort@d . . . 2258, 2266	
	\QS@sort@empty . 2248, 2253	
	\QS@sort@single 2249, 2254	
	\QSinitialize . . . . . . . . 2171, 2283, 2318	
	\QSIr . 2254, 2260, 2268, 2288, 2302, 2307, 2310	
	\QSIrr 2289, 2302, 2303, 2311	
	\QSLr . . . . . 2260, 2267, 2278, 2279, 2287, 2300, 2305, 2309	
	\QSpivotStep . . . . . 2173, 2283, 2287, 2298	
	\QSR . . . . . 2260	
	\QSRr . . . . . 2269, 2290, 2301, 2306, 2312, 2321, 2322, 2323	
	\QSsortStep . . . . . 2175, 2283, 2299, 2300	
	\quad . . . . . 1871	
	<b>R</b>	
	\raisebox . . . . . 1492	
	\relax . . . . . 1555, 2260, 2303, 2305, 2306, 2307, 2415, 2417	
	\renewcommand . . 1242, 1243, 1244, 1245, 1248, 1249, 1263, 1264, 1513, 1514, 1913	
	\repeat . . . . . 2176	
	\RequirePackage . . . . . . . 57, 160, 224, 228, 409, 415, 416,	
		2360, 2816, 2817, 2827
		\right . . . . . 1696
	\Rrightarrow . . . 2710, 2715	
	\Rightarrow . . . 614, 619	
	\rightarrow . . . . 209, 503, 508, 516, 520, 522, 523, 525, 580, 588, 2090, 2593, 2600, 2602, 2605, 2610, 2615, 2616, 2621	
	\rightouterjoin . . . 2342	
	\Roman . . . . . 1247	
	\roman . . . . . 1247, 1249	
	\romannumeral . . . . 2258	
	\rtimes . . . . . 1854	
	\rule 2291, 2313, 2328, 2339	
	<b>S</b>	
	\sb . . . . . 69, 79, 81, 110, 168, 515, 516, 520, 523, 524, 525, 1174, 1176, 1746, 1749, 1752, 1778, 1784, 1933, 2078, 2087, 2615, 2616, 2617, 2622, 2626, 2627, 2630, 2631, 2634, 2635, 2636	
	\scriptscriptstyle . . . . . . . 580, 588, 596	
	\scriptsize . . . 1193, 1432, 1439, 1445, 1507, 1508, 1511, 1512, 2739, 2792, 2822	
	\section . . . . . 48	
	\seq . 1496, 1497, 1498, 1640, 1641, 1642, 1649	
	\setbox . . . . . 2338	
	\setcounter . . . . . 1240, 2292, 2314, 2328	
	\setganttlinklabel . . 1270, 1271, 1272, 1273	
	\setlength . . . . . 2360, 2816, 2817, 2827	



<code>\setmainfont</code> . . . . .	1230	<code>\titlespacing</code> . . . . .	1238	733, 738, 743, 746,
<code>\setmainlanguage</code> . . . .	410	<code>\tl</code> . . . . .	29, 37, 65, 66, 67,	804, 805, 806, 811,
<code>\setminted</code> . . . . .	2487, 2488		68, 69, 70, 73, 74,	812, 841, 871, 898,
<code>\setminus</code> . . . . .	1993		75, 76, 77, 79, 81,	899, 902, 953, 956,
<code>\setsansfont</code> . . . . .	1231		106, 107, 108, 109,	995, 1001, 1002,
<code>\setul</code> . . . . .	2351		110, 111, 112, 115,	1020, 1021, 1022,
<code>\sffamily</code> . . . . .	490,		116, 117, 118, 119,	1041, 1042, 1043, 1044
	1237, 1239, 1341, 2571		120, 121, 164, 165,	<code>\umld</code> . . . . .
<code>\shoveleft</code> . . . . .	1973		166, 167, 168, 169,	<code>\umlHVHagg</code> . . . . .
<code>\shoveright</code> . . . . .	1977		170, 173, 174, 175,	
<code>\Sigma</code> . . . . .	66, 107,		176, 177, 178, 179,	751, 817, 1011
	165, 1143, 1144, 1204		293, 297, 321, 325,	<code>\umlinherit</code> . . . . .
<code>\sigma</code> . . . . .	513, 515, 516		326, 327, 330, 335,	690, 741, 906, 951, 959
<code>\SLASH</code> . . . . .	1561		336, 337, 348, 349,	<code>\umlnote</code> . . . . .
<code>\small</code> . . . . .	1670		350, 351, 362, 368,	692, 908, 1051
<code>\sort</code> . . . . .	2425		371, 374, 383, 397,	<code>\umlreal</code> . . . . .
<code>\sortList</code> . . . . .	2424, 2433		550, 553, 558, 559,	688, 749
<code>\square</code> . . . . .	478		567, 568, 571, 572,	<code>\umlsimpleclass</code> . . . .
<code>\stepcounter</code> 2212, 2217,			1182, 1203, 1204,	648, 649, 653, 655,
2222, 2225, 2227,			1205, 1206, 1209,	656, 657, 683, 836,
2231, 2233, 2237, 2239			1210, 1211, 1212, 1729	837, 838, 897, 949, 950
<code>\str</code> . . . . .	491, 500, 1583,	<code>\tmp</code> . . . . .	1995	<code>\umlstatic</code> . . . . .
	2102, 2111, 2572, 2585	<code>\TmpPlaceEight</code> . . . . .	2050	843, 872
<code>\string</code> . . . . .	1975, 1985	<code>\TmpPlaceFive</code> . . . . .	2047	<code>\umluniagg</code> . . . . .
<code>\StrSubstitute</code> . . . . .	2432, 2434	<code>\TmpPlaceFour</code> . . . . .	2046	904
<code>\strut</code> . . . . .	1800, 1804,	<code>\TmpPlaceNine</code> . . . . .	2051	<code>\umluniassoc</code> . . . . .
	1808, 1812, 1816, 2460	<code>\TmpPlaceOne</code> . . . . .	2043	667,
<code>\subseq</code> 2596, 2629, 2636		<code>\TmpPlaceSeven</code> . . . . .	2049	689, 905, 1029, 1030
		<code>\TmpPlaceSix</code> . . . . .	2048	<code>\umlVHuniassoc</code> . . . .
		<code>\TmpPlaceTen</code> . . . . .	2052	668, 669
		<code>\TmpPlaceThree</code> . . . . .	2045	<code>\umlVHVdep</code> . . . . .
		<code>\TmpPlaceTwo</code> . . . . .	2044	661,
		<code>\TmpScale</code> . . . . .	2063	662, 664, 665, 845, 846
		<code>\TmpTransitionEight</code> . . . . .	2039, 2060	<code>\umlVHVinherit</code> . . . . .
		<code>\TmpTransitionFive</code> . . . . .	2036, 2057	644, 645, 650, 651,
		<code>\TmpTransitionFour</code> . . . . .	2035, 2056	658, 659, 814, 815,
		<code>\TmpTransitionNine</code> . . . . .	2040, 2061	839, 840, 1009, 1010
		<code>\TmpTransitionOne</code> . . . . .	2032, 2053	<code>\umlVHVreal</code> . . . . .
		<code>\TmpTransitionSeven</code> . . . . .	2038, 2059	808, 809, 1046, 1047
		<code>\TmpTransitionSix</code> . . . . .	2037, 2058	<code>\UParrow</code> . . . . .
		<code>\TmpTransitionTen</code> . . . . .	2041, 2062	2712
		<code>\TmpTransitionThree</code> . . . . .	2034, 2055	<code>\url</code> . . . . .
		<code>\TmpTransitionTwo</code> . . . . .	2033, 2054	1681
		<code>\TmpX</code> . . . . .	2064	<code>\usemintedstyle</code> . . . .
		<code>\TmpY</code> . . . . .	2065	2484
		<code>\today</code> . . . . .	1508	<code>\usetikzlibrary</code> . . . .
		<code>\ttfamily</code> . . . . .	2352	58,
				417, 1081, 1285,
				1422, 2029, 2337,
				2378, 2409, 2697, 2785
				V
				<code>\value</code> . . . . .
				2174
				<code>\varepsilon</code> . . . . .
				492,
				503, 504, 1129,
				1746, 1752, 1781, 1792
				<code>\vfill</code> . . . . .
				2460
				<code>\vrule</code> . . . . .
				2667, 2671
				<code>\vspace</code> . . . . .
				1617, 1635, 268

\xintLength	.....	2246		<b>Z</b>		\zustandsnamens@liste
			\ZB	.....	<u>1688</u>	... 1147, 1154, 1155
\xintnthelt	.....	2258	\zB	.....	<u>1687</u>	