

lehramt-informatik

Hermine Bschlangaul *

August 19, 2021

Contents

1	Klassen	3
1.1	Vorlage Theorie-Teil	4
1.2	Vorlage Aufgabensammlung	5
1.3	Vorlage Aufgabe	6
2	Pakete	7
2.1	aufgaben-einbinden.sty	8
2.2	aufgaben-titel.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	Abstrakte Fabrik	31
2.10.2	Adapter	32
2.10.3	Beobachter	33
2.10.4	Dekorierer	34
2.10.5	Einzelstück	35
2.10.6	Erbauer	36
2.10.7	Fabrikmethode	37
2.10.8	Kompositum	38
2.10.9	Modell-Präsentation-Steuerung	39
2.10.10	Zustand	40
2.11	er.sty	42
2.12	formale-sprachen.sty	44
2.13	formatierung.sty	48
2.13.1	Schriftarten / Typographie	48
2.13.2	Farben	48
2.13.3	Überschriften	48
2.13.4	Listen	48
2.13.5	Kasten	48
2.13.6	Header	48

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

2.14	gantt.sty	49
2.15	grafik.sty	50
2.16	graph.sty	51
2.17	hanoi.sty	53
2.18	komplexitaetstheorie.sty	54
2.19	kontrollflussgraph.sty	56
2.20	literatur-dummy.sty	58
2.21	literatur.sty	59
2.22	makros.sty	60
2.23	master-theorem.sty	64
2.24	mathe.sty	68
2.25	minimierung.sty	69
2.26	normalformen.sty	72
2.27	petri.sty	74
2.28	potenzmengen-konstruktion.sty	76
2.29	pseudo.sty	78
2.30	pumping-lemma.sty	79
2.31	quicksort.sty	80
2.32	relationale-algebra.sty	83
2.33	rmodell.sty	84
2.34	sortieren.sty	85
2.35	spalten.sty	87
2.36	struktogramm.sty	88
2.37	syntax.sty	89
2.38	syntaxbaum.sty	91
2.39	synthese-algorithmus.sty	92
2.40	tabelle.sty	95
2.41	typographie.sty	96
2.42	uml.sty	97
2.43	vollstaendige-induktion.sty	99
2.44	wasserfall.sty	101
2.45	wpkalkuel.sty	102

3 Index 103

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-titel.sty

```
19 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
20 \ProvidesPackage{lehramt-informatik-aufgaben-titel}[2020/07/07 Zum
21 Setzen von Aufgaben-Titel.]

22 \ExplSyntaxOn

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

\liSetzeAufgabenTitel Setzen des Titels über eine plist bzw über key-values. Das Makro sollte irgendwann
einmal in \liAufgabenTitel umbenannt werden.
Die Schlüssel-Werte-Paare sind in der Datei basis.sty definiert.

\liSetzeAufgabenTitel{
Titel = Aufgabe 2,
Thematik = Petri-Netz,
Fussnote = sosy:pu:4,
FussnoteSeite = Seite 11,
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\liSetzeAufgabenTitel#1{
26
27 \_setze_variablen_zurueck:
28
29 \tl_clear:N \g_auf_examen_telaufgabe_nr_tl
30
31 \keys_set:nn { aufgabentitel } {
32 #1
33 }
34
35 \_setze_relativen_pfad:
36
37 \tl_if_empty:NTF \g_auf_titel_tl
38 {
39 \msg_fatal:nn { aufgabentitel } { kein-titel }
40 }
41 {
42 }
43
44 \_gib_examen_titel: {}
45
46 \section{\_gib_aufgaben_titel:}
47
48 \bigskip
49 }

50 \ExplSyntaxOff

51
```

2.3 automaten.sty

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

2.3.1 Endlicher Automat

```
54 \liLadePakete{formale-sprachen}

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

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



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

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

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

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

```

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

```

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

```

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

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

```

2.3.2 Kellerautomat

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

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

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

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



```
\liKellerAutomat \liKellerAutomat[\langle automaten-name \rangle]
{\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},
}
```

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

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

```

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

```

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

(a, #: A)
(b, #: ε)

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

```

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

```

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

```

140 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
141     \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
142 }
143 \tikzset{
144     li keller knoten/.style={
145         text width=2cm,
146         align=center,
147         font=\footnotesize,
148     },
149     li kellerautomat/.style={
150         li automat,
151         every edge/.append style={
152             every node/.style={
153                 li keller knoten
154             }
155         }
156     }
157 }

```

2.3.3 Turingmaschine

```

158 \RequirePackage{amssymb}

```

\liTuringLeerzeichen

□

```

159 \def\liTuringLeerzeichen{\Box}

```

```

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

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

```

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

```

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

```

\liTuringUebergangZelle Formatiert einen Zustandsübergang für eine Übergangstabelle.

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

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

```

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

```

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

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

`(z1: \square , L)`

`(\square : \square , R)`

```

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

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

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

\liTuringUeberfuehrung

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

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

```

2.4 basis.sty

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

222 \RequirePackage{xparse}
223 \ExplSyntaxOn

\liLadePakete

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

\liLadeAllePakete

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

Definition einer Komma-getrennten-Liste mit deren Hilfe die vielen globalen Token-List-Variablen definiert werden können.

```
270 \clist_new:N \g_auf_schluesel_clist
271 \clist_set:Nn \g_auf_schluesel_clist {
272   titel,
273   thematik,
```



```

274 fussnote,
275 fussnote_seite,
276 relativer_pfad,
277 examen_nummer,
278 examen_jahr,
279 examen_monat,
280 examen_thema_nr,
281 examen_teilaufgabe_nr,
282 examen_aufgabe_nr,
283 }

```

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

```

284 \clist_map_inline:Nn \g_auf_schluesssel_clist {
285   \tl_new:c {g_auf_#1_tl}
286 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

287 \cs_new:Npn \setze_variablen_zurueck: {
288   \clist_map_inline:Nn \g_auf_schluesssel_clist {
289     \tl_clear:c {g_auf_##1_tl}
290   }
291 }

```

```

292 \keys_define:nn { aufgabentitel }
293 {
294   Titel .tl_gset:N = \g_auf_titel_tl,
295   Thematik .tl_gset:N = \g_auf_thematik_tl,
296   Fussnote .tl_gset:N = \g_auf_fussnote_tl,
297   FussnoteSeite .tl_gset:N = \g_auf_fussnote_seite_tl,
298   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
299   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
300   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
301   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
302   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
303   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_teilaufgabe_nr_tl,
304   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
305 }

```

```

306 \cs_gset:Npn \setze_relativen_pfad: {
307   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
308   {
309     \bool_if:nTF
310     {
311       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
312       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
313       ! \tl_if_empty_p:N \g_auf_examen_monat_tl
314     }
315     {
316       \tl_gset:Nn \g_auf_relativer_pfad_tl {
317         Staatsexamen /
318         \g_auf_examen_nummer_tl /
319         \g_auf_examen_jahr_tl /
320         \g_auf_examen_monat_tl /
321         \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
322         \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl / }
323         \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_tl / }
324       }
325     }
326   } {}
327 }
328 {}
329 }

```

```

330 \cs_gset:Npn \gib_examen_titel: {
331   \cs_set:Nn \trenner: { \, / \, }

```

```

332 \bool_if:nTF
333 {
334   ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
335   ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
336   ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
337   ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
338 }
339 {
340   {
341     \footnotesize
342     \par
343     \noindent
344     Staatsexamen ~
345     \g_auf_examen_nummer_tl \_trenner:
346     \g_auf_examen_jahr_tl \_trenner:
347
348     \tl_case:Nn \g_auf_examen_monat_tl
349     {
350       { 03 } { Frühjahr }
351       { 09 } { Herbst }
352     } \_trenner:
353
354     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
355       Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
356     }
357     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
358       Teilaufgabe ~ Nr. ~ \g_auf_examen_teilaufgabe_nr_tl \_trenner:
359     }
360     \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
361       Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
362     }
363     \par
364     \bigskip
365   }
366 }
367 }

368 \cs_new:Npn \_gib_github_url: {
369   \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
370     Github :~\href{
371       \LehramtInformatikGithubRawDomain /
372       \LehramtInformatikGithubTexRepo /
373       \LehramtInformatikGitBranch /
374       \g_auf_relativer_pfad_tl
375     }{
376       \g_auf_relativer_pfad_tl
377     }
378   }
379 }

380 \cs_new:Npn \_gib_aufgaben_titel: {
381   \g_auf_titel_tl
382
383   \tl_if_empty:NTF \g_auf_thematik_tl
384   {}
385   {
386     \, ~ [
387       \g_auf_thematik_tl
388     ]
389   }
390 }

391 \msg_new:nnn { aufgabentitel } { kein-titel }
392 { Der~Schlüssel~ist~zwingend~notwendig }
393 \ExplSyntaxOff

```

```

\def\LehramtInformatikRepository{/pfad/zum/repository}
394 \input /etc/lehramt-informatik.config.tex
    biblatex not working with lualatex and babel
395 % \RequirePackage{polyglossia}
396 % \setmainlanguage{german}
397
```

2.5 baum.sty

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

401 \RequirePackage{tikz}

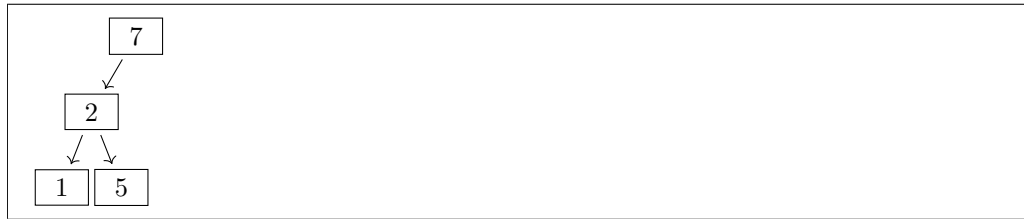
    für li binaer baum
402 \RequirePackage{tikz-qtree}

    Für b baum
403 \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}
```



```
404 \tikzset{
405   li binaer baum/.style={
406     shorten <=2pt,
407     shorten >=2pt,
408     ->,
409     every tree node/.style={
410       minimum width=2em,
411       draw,
412       rectangle
413     },
414     blank/.style={
415       draw=none
416     },
417     edge from parent/.style={
418       draw,
419       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
420     },
421     level distance=1cm,
422     every label/.style={
423       gray,
424       font=\footnotesize,
425       label position=0,
426       label distance=0cm,
427     }
428   },
429 }
```

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}

```



```

430 \tikzset{
431   li bbaum knoten/.style={
432     rectangle split parts=10,
433     rectangle split,
434     rectangle split horizontal,
435     rectangle split ignore empty parts,
436     draw,
437     fill=white
438   },
439   li bbaum/.style={
440     every node/.style={
441       li bbaum knoten
442     },
443     level 1/.style={
444       level distance=12mm,
445       sibling distance=25mm,
446     },
447     every child/.style={
448       shorten <= 2pt,
449       shorten >= 6pt,
450       ->,
451     },
452     level 2/.style={
453       level distance=9mm,
454       sibling distance=15mm,
455     },
456   }
457 }
458

```

2.6 checkbox.sty

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

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

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

465
```


2.7 chomsky-normalform.sty

```

466 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
467 \ProvidesPackage{lehramt-informatik-chomsky-normalform}[2021/03/26
468 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]
469
470 \ExplSyntaxOn
471
472 \liLadePakete{typographie}

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

473 \def\liChomskyUeberschrift#1{
474   {
475     \bfseries
476     \sffamily
477     \str_case:nn {#1} {
478       {1} {Elimination~der~ $\varepsilon$ -Regeln}
479       {2} {Elimination~von~Kettenregeln}
480       {3} {Separation~von~Terminalzeichen}
481       {4} {Elimination~von~mehrelementigen~Nonterminalketten}
482     }
483   }
484 }

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

485 \def\li@chomsky@erklaerung@texte#1{
486   \str_case:nn {#1} {
487     %
488     {1} {
489       Alle~Regeln~der~Form~ $A \rightarrow \varepsilon$ ~werden~eliminiert.~
490       Die~Ersetzung~von~ $A$ ~wird~durch~ $\varepsilon$ ~in~allen~anderen~
491       Regeln~vorweggenommen.
492     }
493     {2} {
494       Jede~Produktion~der~Form~ $A \rightarrow B$ ~mit~ $A, B \in S$ ~wird~
495       als~Kettenregel~bezeichnet.~Diese~tragen~nicht~zur~Produktion~
496       von~Terminalzeichen~bei~und~lassen~sich~ebenfalls~eliminieren.
497     }
498     {3} {
499       Jedes~Terminalzeichen~ $\sigma$ ,~das~in~Kombination~mit~anderen~
500       Symbolen~auftaucht,~wird~durch~ein~neues~Nonterminal~
501        $S_{\sigma}$ ~ersetzt~und~die~Menge~der~Produktionen~durch~die~
502       Regel~ $S_{\sigma} \rightarrow \sigma$ ~ergänzt.
503     }
504     {4} {
505       Alle~Produktionen~der~Form~
506        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ ~
507       werden~in~die~Produktionen~
508        $A \rightarrow$ 
509        $A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow$ 
510        $A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
511        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ ~zerteilt.~
512       Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
513       vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
514     }
515   }
516 }

517 \def\liChomskyErklaerung#1{
518   {
519     \itshape
520     \footnotesize
521     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
522   }

```

523 }

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

524 \def\liChomskyUeberErklaerung#1{

525 \liChomskyUeberschrift{#1}\par

526 \liChomskyErklaerung{#1}

527 }

528 \ExplSyntaxOff

529

2.8 cpm.sty

```

530 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
531 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
532 \RequirePackage{tikz}
533 \liLadePakete{mathe}

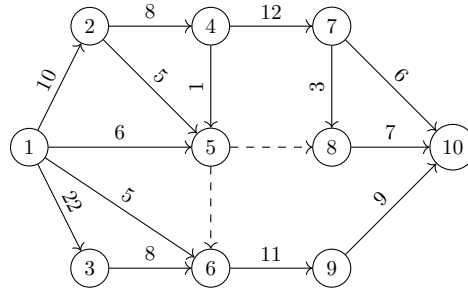
```

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}

```

```

534 \ExplSyntaxOn
535 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
536   \tl_set:Nn \l_name_tl {}
537
538   \keys_define:nn { cpmEreignis } {
539     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
540   }
541
542   \keys_set:nn { cpmEreignis } { #1 }
543
544   \tl_if_empty:NT \l_name_tl {
545     \tl_set:Nn \l_name_tl {#2}
546   }

```

```

547
548 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
549 }
550 \ExplSyntaxOff

\liCpmVorgang liCpmVORGANG\((.*)>(.*))\{(.*)\}
551 \ExplSyntaxOn
552 \NewDocumentCommand { \liCpmVorgang } { 0{ } m m m } {
553   \tl_set:Nn \l_schein_tl {}
554   \tl_set:Nn \l_kritisch_tl {}
555
556   \keys_define:nn { cpmVorgang } {
557     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
558     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
559   }
560
561   \keys_set:nn { cpmVorgang } { #1 }
562
563   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
564 }
565 \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):  $1_{(2 \rightarrow 3)}$ 
566 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
567 \def\liCpmVonZu#1(#2-#3){%
568   \ifmmode%
569     \liCpmVonZuOhneMathe{#1}(#2-#3)%
570   \else%
571     $\liCpmVonZuOhneMathe{#1}(#2-#3)$%
572   \fi%
573 }

```

```

\liCpmVon Let-Abkürzung: \let\v=\liCpmVon
\liCpmVon{1}(2):  $1_{(\rightarrow 2)}$ 
574 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
575 \def\liCpmVon#1(#2){%
576   \ifmmode%
577     \liCpmVonOhneMathe{#1}(#2)%
578   \else%
579     $\liCpmVonOhneMathe{#1}(#2)$%
580   \fi%
581 }

```

```

\liCpmZu Let-Abkürzung: \let\z=\liCpmZu
\liCpmZu{1}(2):  $1_{(\leftarrow 2)}$ 
582 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
583 \def\liCpmZu#1(#2){%
584   \ifmmode%
585     \liCpmZuOhneMathe{#1}(#2)%
586   \else%
587     $\liCpmZuOhneMathe{#1}(#2)$%

```

```

588 \fi%
589 }

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

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

592

```

2.9 cyk-algorithmus.sty

```
593 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
594 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
595 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
596 \def\liKurzeTabellenLinie#1{\ccline{1-#1}}
```

```
\liWortInSprache  $\Rightarrow abc \in L(Y)$ 
597 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
598   \bigskip
599   \noindent
600   $\Rrightarrow$ #1 \in #2$
601 }
```

```
\liWortNichtInSprache  $\Rightarrow abc \notin L(G)$ 
602 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
603   \bigskip
604   \noindent
605   $\Rrightarrow$ #1 \notin #2$
606 }

607
```

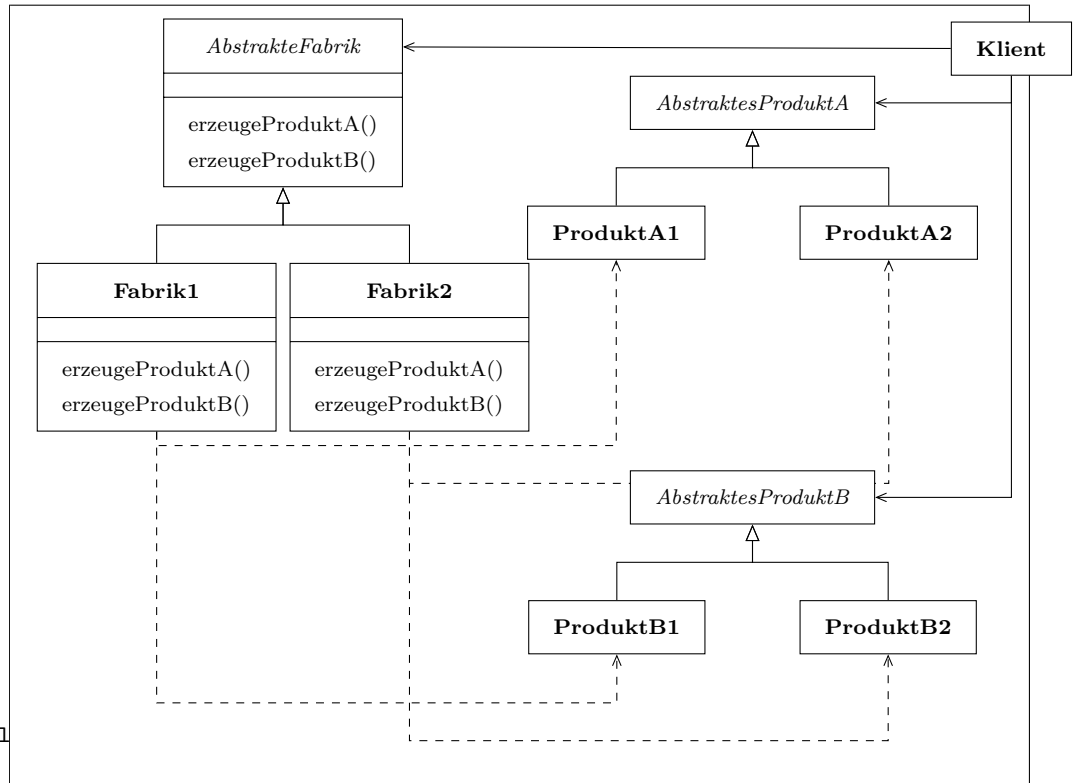
2.10 entwurfsmuster.sty

```

608 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
609 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06 Hilfsmakros
610 zum Setzen von Entwurfsmuster/Design Patterns]
611
612 \RequirePackage{lehramt-informatik-uml}

```

2.10.1 Abstrakte Fabrik



```

613 \def\liEntwurfsAbstrakteFabrikUml{
614   \begin{tikzpicture}
615     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
616       erzeugeProduktA()\n
617       erzeugeProduktB()\n
618     }
619     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{-}{
620       erzeugeProduktA()\n
621       erzeugeProduktB()\n
622     }
623     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{-}{
624       erzeugeProduktA()\n
625       erzeugeProduktB()\n
626     }
627     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
628     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
629
630     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
631     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
632     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
633     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
634     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
635
636     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
637
638     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
639     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
640     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}

```

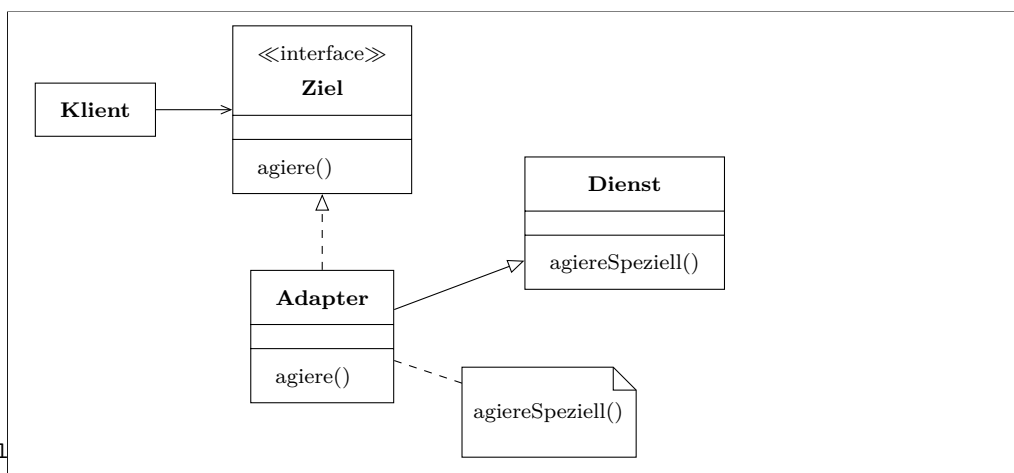
```

641 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
642 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
643
644 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
645 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
646
647 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
648 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
649
650 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
651 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
652 \umlVHuniassoc{Klient}{AbstraktesProduktB}
653 \end{tikzpicture}
654 }

```

2.10.2 Adapter

\liEntwurfsAdapterUml



```

655 \def\liEntwurfsAdapterUml{
656 \begin{tikzpicture}
657 \umlsimpleclass[x=1,y=3]{Klient}{}{}
658 \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
659 \umlclass[x=4,y=0]{Adapter}{}{agiere()}
660 \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
661
662 \umlreal{Adapter}{Ziel}
663 \umluniassoc{Klient}{Ziel}
664 \umlinherit{Adapter}{Dienst}
665
666 \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
667 \end{tikzpicture}
668 \footcite[so ähnlich wie GoF]{wiki:adapter}
669 }

```

\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.

```

670 \def\liEntwurfsAdapterAkteure{
671 \begin{description}
672
673 \item[Ziel (Target)]

```

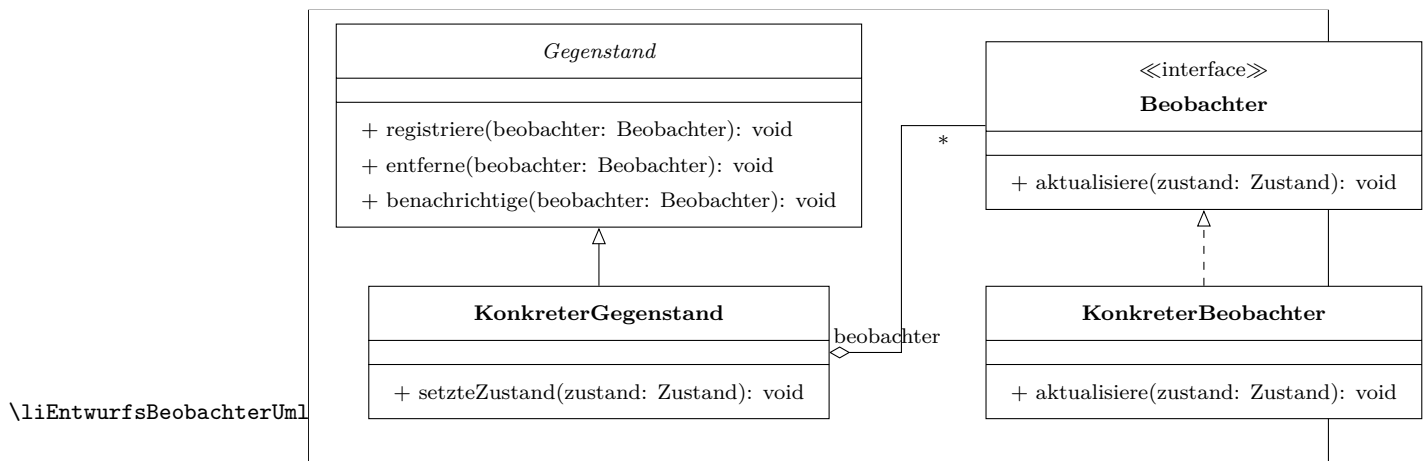


```

674
675 Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
676
677 \item[Klient (Client)]
678
679 Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
680 dabei auf adaptierte Schnittstellen zurück.
681
682 \item[Dienst (Adaptee)]
683
684 Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
685 definierter Schnittstelle an.
686
687 \item[Adapter]
688
689 Der Adapter adaptiert die Schnittstelle des Dienstes auf die
690 Schnittstelle zum Klienten.\footcite{wiki:adapter}
691
692 \end{description}
693 }

```

2.10.3 Beobachter



```

694 \def\liEntwurfsBeobachterUml{
695   \begin{tikzpicture}
696     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
697       + registriere(beobachter: Beobachter): void\\
698       + entferne(beobachter: Beobachter): void\\
699       + benachrichtige(beobachter: Beobachter): void\\
700     }
701     \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
702       + setzteZustand(zustand: Zustand): void
703     }
704     \umlinherit{KonkreterGegenstand}{Gegenstand}
705
706     \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
707       + aktualisiere(zustand: Zustand): void
708     }
709     \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
710       + aktualisiere(zustand: Zustand): void
711     }
712     \umlreal{KonkreterBeobachter}{Beobachter}
713
714     \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
715     {KonkreterGegenstand}{Beobachter}
716   \end{tikzpicture}
717 }

```

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)

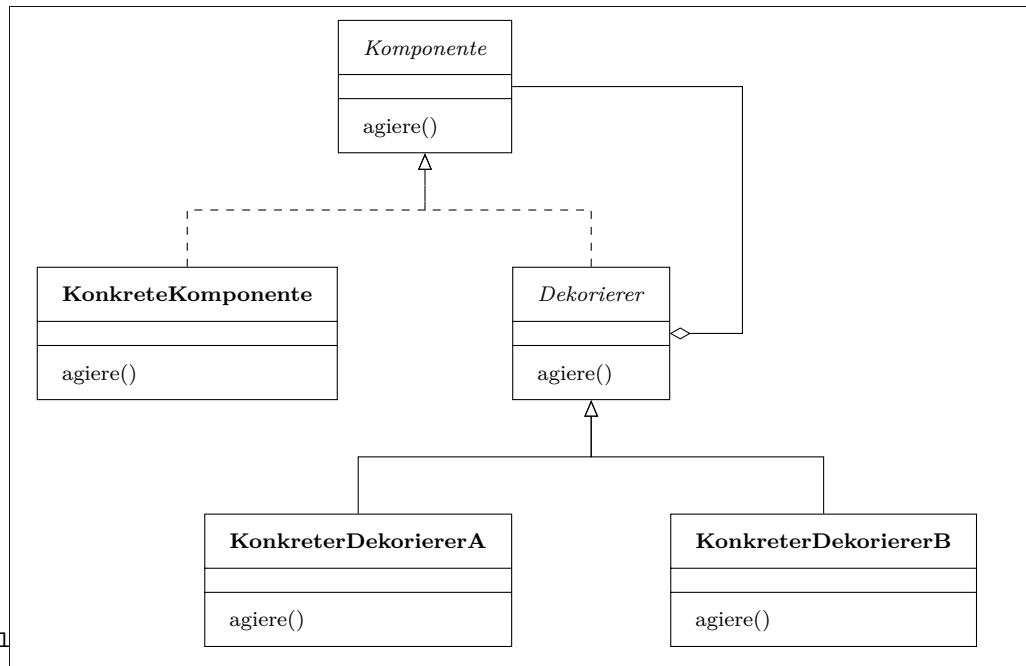
Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert den relevanten Zustand und benachrichtigt alle Beobachter bei Zustandsänderungen über deren Aktualisierungsschnittstelle. Es verfügt über eine Schnittstelle zur Erfragung des aktuellen Zustands.

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.

```

718 \def\liEntwurfsBeobachterAkteure{
719   \begin{description}
720     \item[Gegenstand / Subjekt (Subject / Observable)]
721
722     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
723     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
724     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
725     und Abmeldung von Beobachtern und eine Schnittstelle zur
726     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
727     251]{gof}
728
729     \item[Beobachter (Observer)]
730
731     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
732     genannt) definieren eine Aktualisierungsschnittstelle.
733
734     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
735
736     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
737     den relevanten Zustand und benachrichtigt alle Beobachter bei
738     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
739     verfügt über eine Schnittstelle zur Erfragung des aktuellen
740     Zustands.
741
742     \item[Konkrete Beobachter (ConcreteObserver)]
743
744     Konkrete Beobachter verwalten die Referenz auf ein konkretes
745     Subjekt, dessen Zustand sie beobachten und speichern und dessen
746     Zustand konsistent ist. Sie implementieren eine
747     Aktualisierungsschnittstelle unter Verwendung der
748     Abfrageschnittstelle des konkreten Subjekts.
749     \footcite{wiki:beobachter}
750   \end{description}
751 }
```

2.10.4 Dekorierer



\liEntwurfsDekoriererUml

```

752 \def\liEntwurfsDekoriererUml{
753   \begin{tikzpicture}
754     \umlclass[type=abstract]{Komponente}{-}{agiere()}
755     \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
756     \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
757
758     \umlVHVreal{KonkreteKomponente}{Komponente}
759     \umlVHVreal{Dekorierer}{Komponente}
760
761     \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
762     \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
763
764     \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
765     \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
766
767     \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
768     \footcite{wiki:dekorierer}
769   \end{tikzpicture}
770 }
  
```

2.10.5 Einzelstück

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

771 \def\liEntwurfsEinzelstueckUml{
772   \begin{tikzpicture}
773     \umlclass{Einzelstück}{
774       \umlstatic{- instanz: Einzelstück}\\
775     }{
776       - Einzelstück()\\
777       + gibInstanz(): Einzelstück
778     }
779   \end{tikzpicture}
  
```

780 }

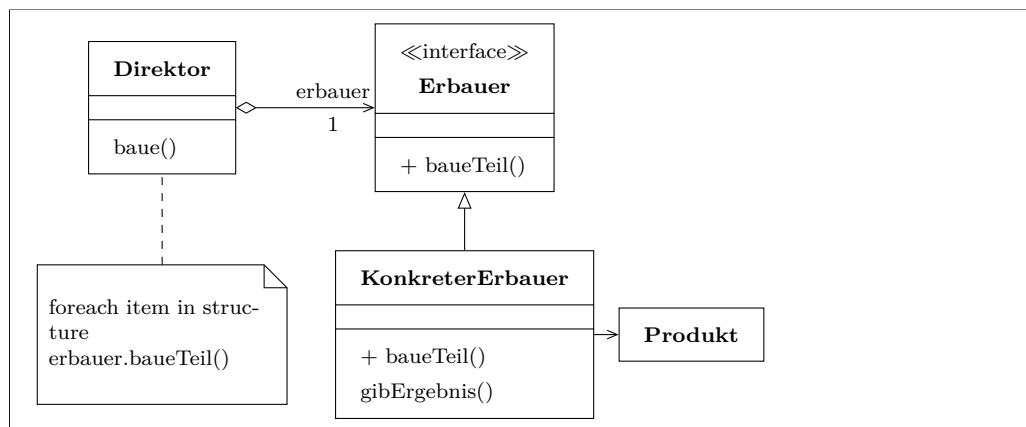
iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

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

```
781 \def\liEntwurfsEinzelstueckAkteure{
782   \begin{description}
783     \item[Einzelstück (Singleton)]
784
785     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
786     nur auf eine einzige Instanz der Klasse zugreifen können.
787   \end{description}
788 }
```

2.10.6 Erbauer

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```
789 \def\liEntwurfsErbauerUml{
790   \begin{tikzpicture}
791     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
792     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
793     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
794       + baueTeil()\
795       gibErgebnis()
796     }
797     \umlclass[x=0,y=3]{Direktor}{}{baue()}
798     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
799     \umluniassoc{KonkreterErbauer}{Produkt}
800     \umlinherit{KonkreterErbauer}{Erbauer}
801
802     \umlnote[x=0,y=0,width=3cm]{Direktor}{
803       foreach item in structure\
804       erbauer.baueTeil()
805     }
806   \end{tikzpicture}
807   \footcite{wiki:erbauer}
808 }
```

\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.

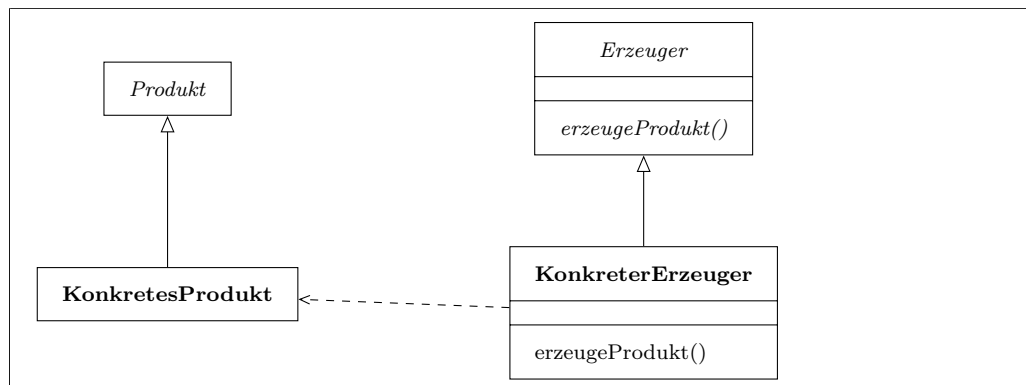
Produkt Das Produkt repräsentiert das zu konstruierende komplexe Objekt.

```

809 \def\liEntwurfsErbauerAkteure{
810   \begin{description}
811     \item[Erbauer]
812
813     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
814     Teile eines komplexen Objektes.
815
816     \item[KonkreterErbauer]
817
818     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
819     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
820     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
821     Schnittstelle zum Auslesen des Produkts.
822
823     \item[Direktor]
824
825     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
826     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
827     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
828     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
829     Klienten.
830
831     \item[Produkt]
832
833     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
834   \footcite{wiki:erbauer}
835 \end{description}
836 }
```

2.10.7 Fabrikmethode

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

837 \def\liEntwurfsFabrikmethodeUml{
838   \begin{tikzpicture}
839     \umlclass[type=abstract]{Produkt}
840     \umlclass[below=2cm of Produkt]{KonkretesProdukt}
841     \umlinherit{KonkretesProdukt}{Produkt}
842
843     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
844       \textit{erzeugeProdukt()}\}
  
```

```

845     }
846     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{}{
847     erzeugeProdukt()
848     }
849     \umlinherit{KonkreterErzeuger}{Erzeuger}
850
851     \umldep{KonkreterErzeuger}{KonkretesProdukt}
852 \end{tikzpicture}
853 }

```

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).

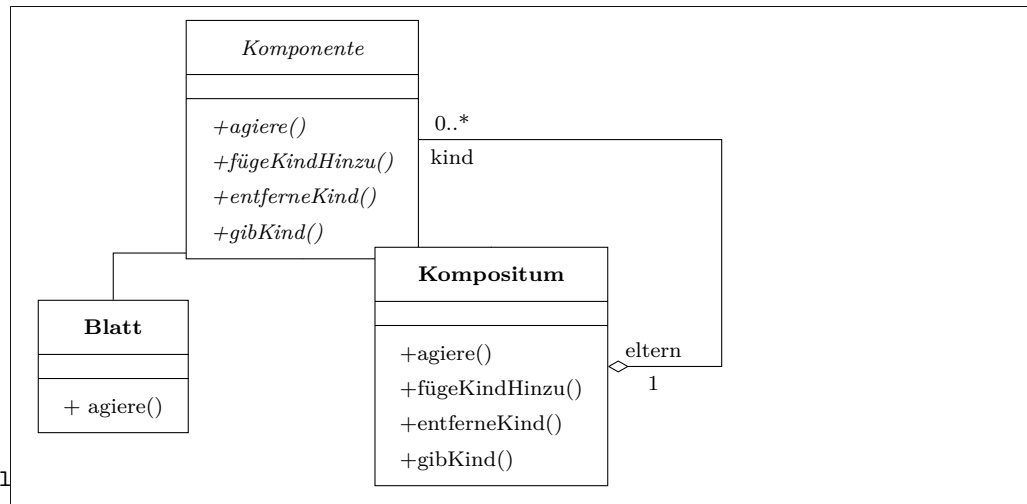
```

854 \def\liEntwurfsFabrikmethodeAkteure{
855   \begin{description}
856     \item[Produkt]
857
858     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
859     zu erzeugende Produkt.
860
861     \item[KonkretesProdukt]
862
863     KonkretesProdukt implementiert die Produkt-Schnittstelle.
864
865     \item[Erzeuger]
866
867     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
868     zu erzeugen und kann eine Default-Implementierung beinhalten.
869
870     \item[KonkreterErzeuger]
871
872     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
873     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
874     Konstruktor einer konkreten Produkt-Klasse aufruft).
875
876     \footcite{wiki:fabrikmethode}
877   \end{description}
878 }
879

```

2.10.8 Kompositum

\liEntwurfsKompositumUml



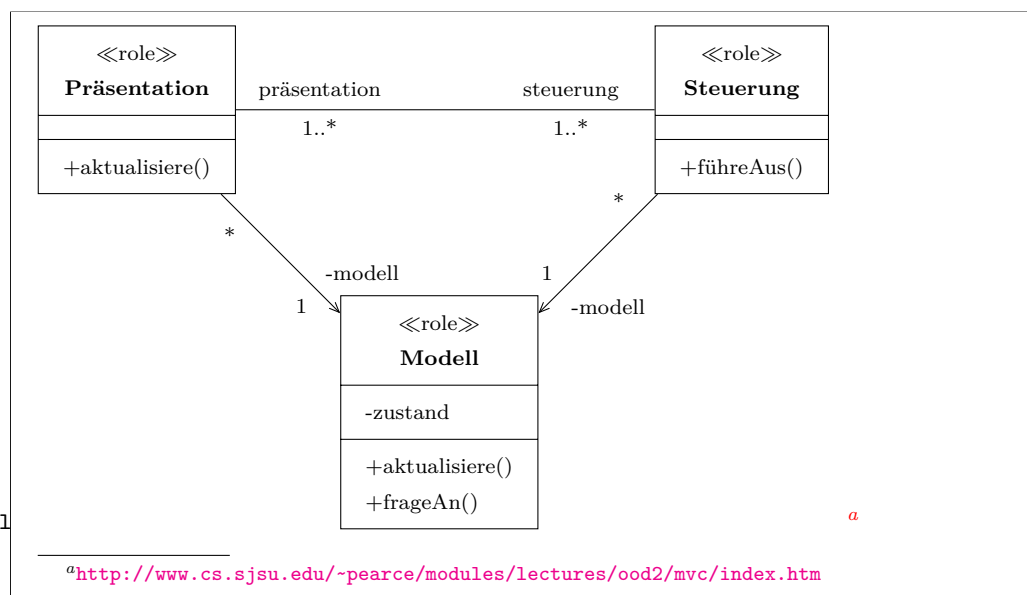
```

880 \def\liEntwurfsKompositumUml{
881   \begin{tikzpicture}
882     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
883       \textit{+agiere()}\
884       \textit{+fügeKindHinzu()}\
885       \textit{+entferneKind()}\
886       \textit{+gibKind()}
887     }
888     \umlclass[x=0]{Blatt}{-}{+ agiere()}
889     \umlclass[x=5]{Kompositum}{-}{
890       +agiere()\
891       +fügeKindHinzu()\
892       +entferneKind()\
893       +gibKind()
894     }
895
896     \umlVHVinherit{Kompositum}{Komponente}
897     \umlVHVinherit{Blatt}{Komponente}
898     \umlHVVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,
899     \end{tikzpicture}
900 }

```

2.10.9 Modell-Präsentation-Steuerung

ModellPraesentationSteuerungUml



^a<http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm>

```

901 \def\liEntwurfsModellPraesentationSteuerungUml{
902   \begin{tikzpicture}

```

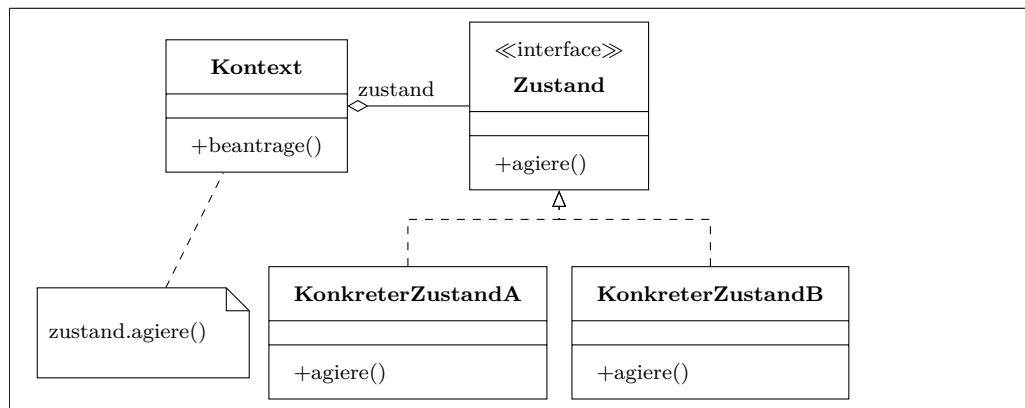
```

903 \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
904 \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
905 \umlclass[y=-4cm,type=role]{Modell}{
906   -zustand
907 }{
908   +aktualisiere()\
909   +frageAn()
910 }
911
912 \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
913 \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
914 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
915 \end{tikzpicture}
916 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
917 }

```

2.10.10 Zustand

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



```

918 \def\liEntwurfsZustandUml{
919   \begin{tikzpicture}
920     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
921     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
922     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
923     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
924
925     \umlVHVreal{KonkreterZustandA}{Zustand}
926     \umlVHVreal{KonkreterZustandB}{Zustand}
927
928     \umlaggreg[arg=zustand,pos=0.4]{Kontext}{Zustand}
929
930     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
931   \end{tikzpicture}
932 }

```

\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.

```

933 \def\liEntwurfsZustandAkteure{
934   \begin{description}
935     \item[Kontext (Context)]

```



```

936
937   definiert die clientseitige Schnittstelle und verwaltet die separaten
938   Zustandsklassen.
939
940   \item[State (Zustand)]
941
942   definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
943   implementiert gegebenenfalls ein Standardverhalten.
944
945   \item[KontreterZustand (ConcreteState)]
946
947   implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
948   verbunden ist.
949 \end{description}
950 }
951

```

2.11 er.sty

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

955 \RequirePackage{tikz-er2}
956 \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}

957 \RequirePackage{soul}
958 \RequirePackage{fontawesome}

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

959 \ExplSyntaxOn

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

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

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

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\e=\liErMpEntity
963 \def\liErMpEntity#1{
964   \liErEntity{#1}
965   \marginpar{
966     \liErEntity{\tiny\faSquareO{}}~E:~#1}
967   }
968 }
□

\liErMpRelationship Let-Abkürzung: \let\r=\liErMpRelationship
969 \def\liErMpRelationship#1{
970   \liErRelationship{#1}
971   \marginpar{
972     \liErRelationship{\tiny\faGg{}}~R:~#1}
973   }
974 }

\liErMpAttribute Let-Abkürzung: \let\a=\liErMpAttribute
975 \def\liErMpAttribute#1{
976   \liErAttribute{#1}
977   \marginpar{
978     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
979   }
980 }

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
datenbank name
981 \def\liErDatenbankName#1{
982   {
983     \footnotesize\texttt{(#1)}
984   }
985 }

986 \ExplSyntaxOff
987

```

2.12 formale-sprachen.sty

```

988 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
989 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
990 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
991
992 \directlua{
993   formale_sprachen = require('lehramt-informatik-formale-sprachen')
994 }
995
996 \RequirePackage{hyperref}
997 \RequirePackage{lehramt-informatik-mathe}

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

998 \def\liMengeOhneMathe#1{\{ #1 \}}
999 \def\liMenge#1{%
1000 \ifmmode%
1001 \liMengeOhneMathe{#1}%
1002 \else%
1003 $\liMengeOhneMathe{#1}$%
1004 \fi%
1005 }

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

1006 \def\liEpsilon{\varepsilon}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung
1007 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1008 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1009 \def\liPotenzmenge#1{\liPotenzmengeOhneMathe{#1}}

\liZustandsmenge \liZustandsmenge{z1, z2}:  $\{z_1, z_2\}$ 
1010 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1011 \let\liZustandsmenge=\liPotenzmenge

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

1012 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1013 \def\liUeberfuehrungsFunktion#1{
1014 \ifmmode
1015 \liUeberfuehrungsFunktionOhneMathe{#1}
1016 \else
1017 $\liUeberfuehrungsFunktionOhneMathe{#1}$
1018 \fi
1019 }

\liAlphabet \liAlphabet{a, b} ( $\Sigma = \{a, b\}$ ):  $\Sigma = \{a, b\}$ 
1020 \def\liAlphabet#1{\Sigma = \{ #1 \}}

\liBandAlphabet \liBandAlphabet{\liTuringLeerzeichen}:  $\Gamma = \Sigma \cup \{\square\}$ 
1021 \def\liBandAlphabet#1{\Gamma = \Sigma \cup \{ #1 \}}

\liZustandsBuchstabe
1022 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross
1023 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1024 \def\zustandsnamens@liste#1#2{
1025   $
1026   \{
1027     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1028   \}
1029   $
1030 }
1031 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1032 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1033 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1034 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1035 \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}

1036 \NewDocumentEnvironment { liProduktionsRegeln }
1037 { 0{P} +b }
1038 {
1039   \noindent
1040   $#1 = \{ $
1041   \vspace{-0.2cm}
1042   \begin{align*}
1043     \directlua{formale_sprachen.produktions_regeln('#2')}
1044   \end{align*}
1045   \vspace{-1.5cm}
1046   \begin{flushright}\}\end{flushright}
1047 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1048 \def\liProduktionen#1{
1049   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1050 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1051 \def\liZustandsnameTiefgestellt#1{
1052   \ifmmode
1053     \liZustandsBuchstabe\sb{#1}
1054   \else
1055     $\liZustandsBuchstabe\sb{#1}$
1056   \fi
1057 }

1058 \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}

1059 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1060   $
1061   \tl_if_empty:nTF {#1} {} {#1 =}
1062   \{
1063     \, #2 \,
1064     |
1065     \, #3 \,
1066   \}$
1067 }
1068 \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

1069 \def\liFlaci#1{%
1070   \par
1071   {%
1072     \scriptsize
1073     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1074     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1075     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1076     \href{https://flaci.com/#1}{flaci.com/#1}%
1077   }%
1078   \par
1079 }

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

1080 \ExplSyntaxOn
1081 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1082   \tl_set:Nn \l_variablen_tl {V}
1083   \tl_set:Nn \l_alphabet_tl {\Sigma}
1084   \tl_set:Nn \l_produktionen_tl {P}
1085   \tl_set:Nn \l_start_tl {S}
1086
1087   \keys_define:nn { grammatik } {
1088     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1089     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1090     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1091     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1092   }
1093
1094   \keys_set:nn { grammatik } { #2 }
1095
1096   $#1 = (
1097     \l_variablen_tl,
1098     \l_alphabet_tl,
1099     \l_produktionen_tl,
1100     \l_start_tl
1101   )$

```

```
1102 }  
1103 \ExplSyntaxOff  
  
1104
```

2.13 formatierung.sty

```
1105 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1106 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.13.1 Schriftarten / Typographie

```
1107 \RequirePackage{mathpazo}
1108 \RequirePackage[no-math]{fontspec}
1109 \setmainfont{texgyrepagella}
1110 \setsansfont{QTAncientOlive}
1111 \RequirePackage{sectsty}
1112 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.13.2 Farben

```
1113 \RequirePackage{xcolor}
1114 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.13.3 Überschriften

```
1115 \RequirePackage{titlesec}
1116 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1117 \titlespacing{\chapter}{0pt}{0pt}{*1}
1118 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1119 \setcounter{secnumdepth}{0}
```

2.13.4 Listen

```
1120 \RequirePackage{paralist}
1121 \renewcommand\labelitemi{-}
1122 \renewcommand\labelitemii{-}
1123 \renewcommand\labelitemiii{-}
1124 \renewcommand\labelitemiv{-}
1125 % Counter: enumi enumii enumiii enumiv
1126 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1127 \renewcommand{\labelenumi}{(\alph{enumi})}
1128 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.13.5 Kasten

```
1129 \RequirePackage{mdframed}
1130 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1131 \NewDocumentEnvironment { liKasten }{ } {
1132   \begin{mdframed}[backgroundcolor=white!0]
1133 } {
1134   \end{mdframed}
1135 }
```

2.13.6 Header

```
1136 \RequirePackage{fancyhdr}
1137 \fancyhead[L,C,R]{}
1138 \fancyfoot[L]{}
1139 \fancyfoot[C]{}
1140 \fancyfoot[R]{\thepage}
1141 \pagestyle{fancy}
1142 \renewcommand{\headrulewidth}{0pt}
1143 \renewcommand{\footrulewidth}{0pt}
1144
```


2.14 gantt.sty

```

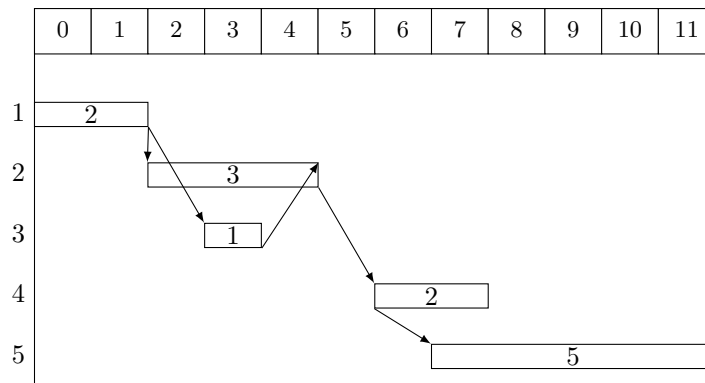
1145 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1146 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1147 \RequirePackage{tikz-uml}
1148 \RequirePackage{pgfgantt}
1149 \setganttlinklabel{f-s}{}
1150 \setganttlinklabel{s-s}{}
1151 \setganttlinklabel{f-f}{}
1152 \setganttlinklabel{s-f}{}
1153

```

2.15 grafik.sty

```
1154 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1155 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1156 TikZ, um Grafiken zeichnen zu können.]
1157 \RequirePackage{tikz}
1158
```

2.16 graph.sty

```

1159 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1160 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1161 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1162 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1163 \RequirePackage{blkarray}
1164 \usetikzlibrary{arrows.meta}

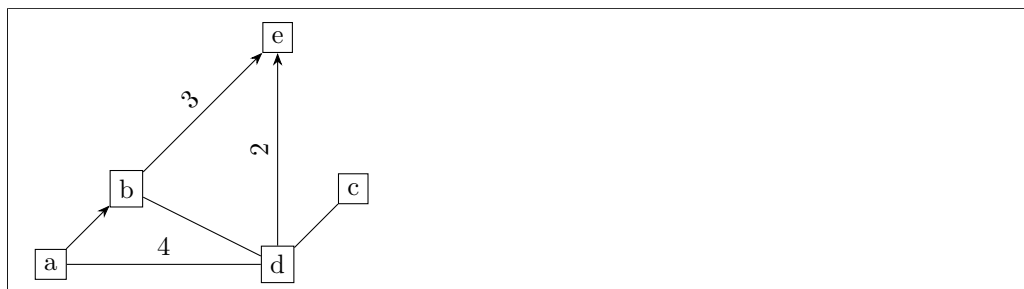
```

```

\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}

```



```

1165 \tikzset{
1166   li graph/.style={
1167     every node/.style={
1168       rectangle,
1169       draw,
1170     },
1171     every edge/.style={
1172       >={Stealth[black]},
1173       draw,
1174     },
1175     every edge/.append style={
1176       every node/.style={
1177         sloped,
1178         auto,
1179       }
1180     }
1181   },
1182   li markierung/.style={
1183     ultra thick,
1184   }
1185 }

```

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}

```

```

1186 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1187

```

2.17 hanoi.sty

```
1188 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1189 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1190 von Hanoi-Grafiken]
```

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

```
1191 \RequirePackage{tikz}
1192 \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}

1193 \def\li@mset #1[#2]=#3{%
1194 \expandafter\xdef\csname #1#2\endcsname{#3}
1195 }
1196 \def\li@mget #1[#2]{%
1197 \csname #1#2\endcsname
1198 }
1199 \def\li@minc #1[#2]+=#3{%
1200 \pgfmathparse{\li@mget #1[#2]+#3}%
1201 \li@mset #1[#2]=\pgfmathresult
1202 }
1203
1204 \def\liHanoi#1#2{
1205   \edef\li@numdiscs{#1}
1206   \def\li@sequence{#2}
1207   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1208     % init colors
1209     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1210     \li@mset col[\j]={\c};
1211     % draw poles and init pole counters
1212     \foreach \j in {1,2,3}{
1213       \li@mset pos[\j]=0
1214       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1215     }
1216     % draw base
1217     \draw (.5,-.5) -- +(3,0);
1218     % draw discs
1219     \foreach[count=\k] \i/\j in \li@sequence{
1220       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1221       \li@minc pos[\j]+=.5}
1222     }
1223   \end{tikzpicture}
1224 }

1225
```

2.18 komplexitaetstheorie.sty

```

1226 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1227 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1228 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1229 Polynomialzeitreduktion.]

```

Let-Abkürzungen

```

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

```

```

1230 \liLadePakete{mathe}

```

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

```

1231 \RequirePackage{mdframed}

```

```

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

```

```

1232 \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

```

```

1233 \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`

```

1234 \def\liProblemBeschreibung#1#2#3{
1235   \begin{mdframed}[
1236     userdefinedwidth=9cm,
1237     align=center,
1238     backgroundcolor=white!0,
1239   ]
1240     \centerline{\large\liProblemName{#1}}
1241
1242     \medskip
1243
1244     \begin{description}
1245       \item[Gegeben:] #2
1246       \item[Frage:] #3
1247     \end{description}
1248   \end{mdframed}
1249 }

```

```

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

1250 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1251 \begin{displaymath}
1252 \liProblemName{#1}
1253 \preceq_{#2}
1254 \liProblemName{#3}
1255 \end{displaymath}
1256 }

\liProblemVertexCover

1257 \def\liProblemClique{%
1258 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1259 Mindestgröße  $n$  in einem gegebenen Graphen.
1260 \footcite[Seite 76]{theo:fs:4}
1261 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1262 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1263 \footcite{wiki:cliquenproblem}
1264 }

\liProblemVertexCover

1265 \def\liProblemVertexCover{%
1266 %
1267 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1268 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1269 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1270 \footcite{wiki:knotenueberdeckung}
1271
1272 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1273 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1274 mindestens einem Knoten aus  $U$  verbunden ist.
1275 \footcite[Seite 78]{theo:fs:4}%
1276 }

\liProblemSubsetSum Kein Karp-21-Problem

1277 \def\liProblemSubsetSum{%
1278 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1279 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1280 \footcite{wiki:teilsommenproblem}
1281 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1282 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1283 größer als eine gegebene obere Schranke  $c$  ist.
1284 \footcite[Seite 74]{theo:fs:4}
1285 }

\liProblemSubsetSum Kein Karp-21-Problem

1286 \def\liProblemSat{%
1287 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1288 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1289 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1290 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1291 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1292 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1293 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1294 aufgestellt werden.
1295 \footcite[Seite 71]{theo:fs:4}
1296 }

1297

```

2.19 kontrollflussgraph.sty

```
1298 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1299 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1300 \RequirePackage{tikz}
1301 \usetikzlibrary{positioning}
1302 \tikzset{
1303   li kontrollfluss/.style={
1304     knoten/.style={
1305       circle,
1306       draw
1307     },
1308     usebox/.style={
1309       draw,
1310       rectangle,
1311       font=\scriptsize,
1312       anchor=west,
1313       align=left,
1314     },
1315     bedingung/.style={
1316       midway,
1317       draw=none,
1318       font=\scriptsize
1319     },
1320     knotenbeschriftung/.style={
1321       draw,
1322       rectangle,
1323       midway,
1324       font=\scriptsize
1325     },
1326     wahr/.style={
1327       thick
1328     },
1329     falsch/.style={
1330       dashed
1331     },
1332     every node/.style={
1333       circle,
1334       draw,
1335     },
1336     every edge/.append style={
1337       every node/.style={
1338         draw=none,
1339         bedingung,
1340       }
1341     },
1342     every path/.style={
1343       draw,
1344       ->,
1345     },
1346     every pin/.style={
1347       draw,
1348       dotted,
1349       rectangle,
1350       pin position=right
1351     },
1352     every pin edge/.style={
1353       dotted,
1354       arrows=-,
1355     }
1356   }
1357 }
```

liKontrollflussgraph


```

1358 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1359   \begin{tikzpicture}[
1360     li kontrollfluss,
1361     #1
1362   ]
1363 } {
1364   \end{tikzpicture}
1365 }

\liAnweisung
1366 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1367 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1368 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1369 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1370 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1371 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1372 \ExplSyntaxOn
1373 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1374 {
1375   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1376   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1377   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1378 }
1379 \ExplSyntaxOff

1380

```

2.20 literatur-dummy.sty

```
1381 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1382 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1383 \def\literatur{}

\footcite

1384 % \RequirePackage[stable,multiple]{footmisc}
1385 \NewDocumentCommand{ \footcite } { o m }{}

1386
```

2.21 literatur.sty

```
1387 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1388 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1389 \RequirePackage{csquotes}
1390 \RequirePackage[
1391   bibencoding=utf8,
1392   citestyle=authortitle,
1393   backend=biber,
1394 ]{biblatex}
1395 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1396 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1397 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1398 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1399 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1400 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1401 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1402 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1403 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1404 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1405 % To allow footnotes in the heading
1406 \RequirePackage[stable,multiple]{footmisc}

\literatur

1407 \def\literatur{\printbibliography[heading=subbibliography]}

1408
```

2.22 makros.sty

```
1409 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1410 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1411 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1412 anderen Paket passen]
1413 \RequirePackage{hyperref}
1414 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1415 \RequirePackage{paralist}

\inhaltsverzeichnis
1416 \def\inhaltsverzeichnis {
1417   \begin{mdframed}
1418     \begin{group}
1419       \let\clearpage\relax
1420       \tableofcontents
1421     \end{group}
1422   \end{mdframed}
1423 }

\memph \mephm (\marginpar and \emph)
1424 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1425 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1426 \newcommand{\liPseudoUeberschrift}[1]{
1427   \bigskip
1428   \noindent
1429   \textsf{\textbf{#1}}
1430   \noindent
1431 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1432 \newcommand{\liBeschriftung}[1]{
1433   \par
1434   \noindent
1435   \medskip
1436   \textbf{#1}:
1437   \medskip
1438   \noindent
1439 }

\hinweis
1440 \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.
1441 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1442 \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.
1443 \RequirePackage{xparse}
1444 \ExplSyntaxOn
```

```

1445 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1446 {
1447   \str_case:nn {#1} {
1448     {standard} {
1449       \def\beschriftung{}
1450       \mdfsetup{backgroundcolor=gray!5,linecolor=gray}
1451     }
1452     {richtig} {
1453       \def\beschriftung{richtig}
1454       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1455     }
1456     {falsch} {
1457       \def\beschriftung{falsch}
1458       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1459     }
1460     {muster} {
1461       \def\beschriftung{Musterlösung}
1462       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1463     }
1464   }
1465   \ifx\beschriftung\empty\else
1466     \noindent
1467     \textbf{\beschriftung{:}}
1468     \fi
1469     \begin{mdframed}
1470   }
1471 {\end{mdframed}}

```

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

```

1472 \NewDocumentEnvironment{ liAdditum } { o }
1473 {
1474   \begin{mdframed}[backgroundcolor=yellow!5]
1475     \IfNoValueTF {#1}
1476     { \liPseudoUeberschrift{Additum} }
1477     { \liPseudoUeberschrift{Additum:~#1} }
1478   }
1479 {\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.

```

1480 \NewDocumentEnvironment{ liExkurs }{o +b}{
1481   \vspace{0.2cm}%
1482   \begin{mdframed}[
1483     backgroundcolor=white,
1484     bottomline=false,
1485     innermargin=1cm,
1486     leftline=true,
1487     linecolor=black,
1488     linewidth=0.1cm,
1489     outermargin=1cm,
1490     rightline=false,
1491     topline=false,
1492   ]

```

```

1493 \footnotesize
1494 \noindent%
1495 \textbf{Exkurs:~#1}\par%
1496 \noindent%
1497 #2
1498 \end{mdframed}
1499 \vspace{0.2cm}
1500 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1501 \cs_new:Npn \listen@punkt #1 {\item #1}
1502 \NewDocumentEnvironment { liQuellen }{ +b }
1503 {
1504   \seq_clear_new:N \l_quellen
1505   \seq_set_split:Nnn \l_quellen {\item} {#1}
1506   \seq_remove_all:Nn \l_quellen {}
1507   \begin{mdframed}[backgroundcolor=white!0]
1508     \footnotesize
1509     \noindent
1510     \textsf{\textbf{Weiterführende~Literatur:}}
1511     \medskip
1512     \begin{compactitem}
1513       \seq_map_function:NN \l_quellen {\listen@punkt}
1514     \end{compactitem}
1515   \end{mdframed}
1516   %
1517   \makeatletter
1518   \par\@afterindentfalse\@afterheading
1519   \makeatother
1520 } {}

```

liLernkartei

```

1521 \NewDocumentEnvironment { liLernkartei }{ m +b }
1522 {
1523   \begin{mdframed}
1524     \footnotesize
1525     \noindent%
1526     \textbf{Lernkarteikarte:~#1}\par%
1527     \noindent%
1528     #2
1529   \end{mdframed}
1530 } {}

```

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

```

1531 \NewDocumentEnvironment { liDiagramm }{ m +b }
1532 {
1533   \begin{mdframed}[backgroundcolor=white!0]
1534     \small
1535     \noindent%
1536     \textit{#1}:
1537     \begin{center}

```

```

1538 #2
1539 \medskip
1540 \end{center}
1541 \end{mdframed}
1542 } {}
1543 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1544 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1545 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1)}}
1546 }
1547

\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.
1548 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1549 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1)}}
1550 }

\zB
1551 \def\zB{z.\,B. }

\ZB
1552 \def\ZB{Z.\,B. }

\dh
1553 \def\dh{d.\,h. }
1554

```

2.23 master-theorem.sty

1555 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1556 \ProvidesPackage{lehramt-informatik-master-theorem}[2021/04/13]

Let-Abkürzungen

\let\0=\li0

\let\o=\li0mega

\let\T=\liT

\let\t=\liTheta

\liMasterVariablenDeklaration

{3} % a

{3} % b

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

\liMasterFallRechnung

% 1. Fall

{für $\varepsilon = 4$: \}

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

% 2. Fall

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

% 3. Fall

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

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

\liMasterVariablenDeklaration

{ } % a

{ } % b

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

\liMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

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

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

1557 \ExplSyntaxOn

1558 \RequirePackage{amsmath}

\liRundeKlammer

1559 \def\liRundeKlammer#1{

1560 \negthinspace \left(#1 \right)

1561 }

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

1562 \def\liThetaOhneMathe#1{

1563 \Theta \liRundeKlammer{#1}

1564 }

1565 \def\liTheta#1{

1566 \ifmmode

1567 \liThetaOhneMathe{#1}

1568 \else

1569 \$\liThetaOhneMathe{#1}\$

1570 \fi

1571 }


```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1572 \def\liOmegaOhneMathe#1{
1573 \Omega \liRundeKlammer{#1}
1574 }
1575 \def\liOmega#1{
1576 \ifmmode
1577 \liOmegaOhneMathe{#1}
1578 \else
1579 $\liOmegaOhneMathe{#1}$
1580 \fi
1581 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1582 \def\liOOhneMathe#1{
1583 \mathcal{O} \liRundeKlammer{#1}
1584 }
1585 \def\liO#1{
1586 \ifmmode
1587 \liOOhneMathe{#1}
1588 \else
1589 $\liOOhneMathe{#1}$
1590 \fi
1591 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1592 \def\liTOhneMathe#1#2{
1593 \tl_if_blank:nTF {#1}
1594 {}
1595 {#1 \cdot }
1596 T
1597 \liRundeKlammer{\frac{n}{#2}}
1598 }
1599 \def\liT#1#2{
1600 \ifmmode
1601 \liTOhneMathe{#1}{#2}
1602 \else
1603 $\liTOhneMathe{#1}{#2}$
1604 \fi
1605 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1606 \def\liRekursionsGleichung{
1607 $T(n) = \liT{a}{b} + f(n)$
1608 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1609 \def\liBedingungEins{
1610 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1611 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1612 \def\liBedingungZwei{
1613 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1614 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1615 \def\liBedingungDrei{
1616 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1617 }

1618 \ExplSyntaxOff

```

\liMasterVariablen

```

1619 \def\liMasterVariablen{
1620   \begin{displaymath}
1621     T(n) = \liT{a}{b} + f(n)
1622   \end{displaymath}
1623
1624   \begin{itemize}
1625     \item[$a = $]
1626       Anzahl der Unterprobleme in der Rekursion
1627
1628     \item[$\textstyle{\frac{1}{b}} = $]
1629       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1630       repräsentiert wird
1631
1632     \item[$f(n) = $]
1633       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1634       die Kombination der Teillösungen entstehen
1635   \end{itemize}
1636   \footcite{wiki:master-theorem}
1637   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1638 }

```

\liMasterFaelle

```

1639 \def\liMasterFaelle{
1640   \begin{description}
1641     \item[1. Fall:]
1642        $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1643
1644       \hfill falls \liBedingungEins
1645       für  $\varepsilon > 0$ 
1646
1647     \item[2. Fall:]
1648        $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1649
1650       \hfill falls \liBedingungZwei
1651
1652     \item[3. Fall:]
1653        $T(n) \in \liTheta{f(n)}$ 
1654
1655       \hfill falls \liBedingungDrei
1656       für  $\varepsilon > 0$ 
1657       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1658       gilt:
1659        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1660     \end{description}
1661 }

```

\liMasterVariablenDeklaration

```

1662 \def\liMasterVariablenDeklaration#1#2#3{
1663   \begin{description}
1664     \item[Allgemeine Rekursionsgleichung:] \strut
1665
1666     \liRekursionsGleichung
1667
1668     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1669
1670     #1
1671
1672     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1673
1674     um  $\frac{1}{b}$  also  $b = \#2$ 
1675
1676     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut

```

```

1677
1678     $#3$
1679
1680     \item[Ergibt folgende Rekursionsgleichung:] \strut
1681
1682      $T(n) = T_{\#1\#2} + \#3$ 
1683 \end{description}
1684 }

\liMasterFallRechnung
1685 \def\liMasterFallRechnung#1#2#3{
1686 \begin{description}
1687 \item[1. Fall:] \liBedingungEins:
1688
1689     #1
1690
1691 \item[2. Fall:] \liBedingungZwei:
1692
1693     #2
1694
1695 \item[3. Fall:] \liBedingungDrei:
1696
1697     #3
1698 \end{description}
1699 }

\liMasterExkurs
1700 \def\liMasterExkurs{
1701 \begin{liExkurs}[Master-Theorem]
1702 \liMasterVariablen
1703
1704 \noindent
1705 Dann gilt:
1706
1707 \liMasterFaelle
1708 \end{liExkurs}
1709 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1710 \def\liMasterWolframLink#1{
1711 Berechne die Rekursionsgleichung auf WolframAlpha:
1712 \href{https://www.wolframalpha.com/input/?i=\#1\{WolframAlpha\}}
1713 }

1714

```

2.24 mathe.sty

```
1715 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1716 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1717
1718 % for example \ltimes \rtimes
1719 %\RequirePackage{amssymb}
1720 \RequirePackage{amsmath}
1721
1722 %%
1723 % \mlq \mrq
1724 %%
1725 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1726 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1727
```

2.25 minimierung.sty

```

1728 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1729 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1730 Minimierungsalgorithmus von einem NEA zu einem DEA]

1731 \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

1732 \def\liFussnote#1{\$x_{#1}\$}

1733 \def\li@fussnote@text#1#2{
1734   \liFussnote{#1}
1735   \quad
1736   {\footnotesize #2}
1737 }

\liFussnoteEinsText
1738 \def\liFussnoteEinsText{
1739   \li@fussnote@text{1}
1740   {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1741 }

\liFussnoteZweiText
1742 \def\liFussnoteZweiText{
1743   \li@fussnote@text{2}
1744   {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1745 }

\liFussnoteDreiText
1746 \def\liFussnoteDreiText{
1747   \li@fussnote@text{3}

```

```

1748 {In weiteren Iterationen markierte Zustände.}
1749 }

\liFussnoteVierText
1750 \def\liFussnoteVierText{
1751   \li@fussnote@text{4}
1752   {...}
1753 }

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



1754 \def\liFussnoten{
1755   \bigskip
1756
1757   \noindent
1758   \liFussnoteEinsText
1759
1760   \noindent
1761   \liFussnoteZweiText
1762
1763   \noindent
1764   \liFussnoteDreiText
1765
1766   \noindent
1767   \liFussnoteVierText
1768 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1769 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1770 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1771 \def\liZustandsPaar#1#2{
1772   $(
1773     \liZustandsPaarVariablenName_#1,
1774     \liZustandsPaarVariablenName_#2
1775   )$
1776 }

liUebergangsTabelle
1777 \renewcommand{\arraystretch}{1.4}
1778 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1779   \liPseudoUeberschrift{Übergangstabelle}
1780   \begin{center}
1781     \begin{tabular}{r|l|l}
1782       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
1783     \end{tabular}
1784   \end{center}
1785 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1787 \ExplSyntaxOn
1788 \def\liUeberschriftDreiecksTabelle{
1789   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1790 }

```

\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. —

```

1791 \def\liMinimierungErklaerung{
1792   %\footcite[Seite~19]{koenig}
1793   \liParagraphMitLinien{
1794     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1795     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1796     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1797     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1798      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1799     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1800     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1801     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1802     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1803   }
1804 }
1805 \ExplSyntaxOff
1806

```

2.26 normalformen.sty

```

1807 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1808 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
1809 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1810 Attributhülle]
1811 \liLadePakete{mathe}
1812 \directlua{
1813   helfer = require('lehramt-informatik-helfer')
1814   normalformen = require('lehramt-informatik-normalformen')
1815 }

```

Let-Abkürzungen

```

\let\ah=\liAttributHuelle
\let\ahL=\liAttributHuelleLinksReduktion
\let\fa=\liFunktionaleAbhaengigkeit
\let\FA=\liFunktionaleAbhaengigkeiten
\let\m=\liAttributMenge
\let\r=\liRelation

```

```

1816 \def\liTeilen#1{
1817   \directlua{tex.print(normalformen.teilen('#1'))}
1818 }

```

```

\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  Regulärer Ausdruck zum Konvertieren AttrHülle\((.*)\) \\\ah{$1}
1819 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
1820 \def\liAttributHuelle#1{
1821   \ifmmode
1822     \liAttributHuelleOhneMathe{\#1}
1823   \else
1824     $\liAttributHuelleOhneMathe{\#1}$
1825   \fi
1826 }

```

```

\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
1827 \def\liAttributMenge#1{\{ \textit{\#1} \}}

```

liAHuelle

```

1828 \NewDocumentEnvironment{ liAHuelle } { +b } {
1829   \begingroup
1830   \footnotesize
1831   \begin{multline*}
1832     \#1
1833   \end{multline*}
1834   \endgroup
1835 } { }

```

AttributHuelleLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multiline

```

Let-Abkürzung: \let\ahL=\liAttributHuelleLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
1836 \def\liAttributHuelleLinksReduktion#1#2#3{
1837   \shoveleft{
1838     \liAttributHuelleOhneMathe{FA, \liAttributMenge{\liAttributMenge{\#1} - \liAttributMenge{\#2}}
1839   } \\\
1840   \shoveright{
1841     \liAttributMenge{\#3}
1842   } \\\
1843 }

```



```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \liFunktionaleAbhaengigkeit{A, B -> C, D} Regulärer Ausdruck zum Konvertieren:
    \$(.*) \rightarrow (.*)\$ \fa{$1 -> $2}
1844 \def\liFunktionaleAbhaengigkeit#1{%
1845     \directlua{normalformen.drucke_funk_abhaengigkeit('#1')}%
1846 }

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

Let-Abkürzung: \let\FA=\liFunktionaleAbhaengigkeiten
    Regulärer Ausdruck zum Konvertieren: \item \$(.*) \rightarrow (.*)\$ \$(.*) \rightarrow
    $1 -> $2;
1847 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
1848     \par
1849     \noindent
1850     #1 $= \{$
1851     \par
1852     \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
1853     \par
1854     \noindent$\}$
1855 }

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

```

$$R_3(A, B, C)$$

```

    Regulärer Ausdruck zum Konvertieren: \$(R.*)\((.*)\)\$ \liRelation[$1]{$2}
1856 \NewDocumentCommand {\liRelation} { O{R} m } {
1857     $\directlua{
1858         local name = helper.konvertiere_tiefgestellt('#1')
1859         tex.print(name)
1860     }\textit{\,#2\,}
1861 }
1862

```

2.27 petri.sty

```
1863 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1864 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]
```

Let-Abkürzungen

```
\let\t=\liPetriTransitionsName
\let\tp=\liPetriTransPfeile
\let\k=\liPetriErreichKnotenDrei
```

```
1865 \RequirePackage{tikz}
1866 \usetikzlibrary{petri,arrows.meta}
```

Für die Darstellungsmatrix

```
1867 \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

```
1868 \def\liPetriSetzeSchluessel{%
1869   \def\TmpTransitionOne{}%
1870   \def\TmpTransitionTwo{}%
1871   \def\TmpTransitionThree{}%
1872   \def\TmpTransitionFour{}%
1873   \def\TmpTransitionFive{}%
1874   \def\TmpTransitionSix{}%
1875   \def\TmpTransitionSeven{}%
1876   \def\TmpTransitionEight{}%
1877   \def\TmpTransitionNine{}%
1878   \def\TmpTransitionTen{}%
1879   \pgfkeys{/petri/.cd,
1880     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
1881     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
1882     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
1883     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
1884     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

1885 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
1886 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
1887 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
1888 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
1889 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
1890 t1/.store in=\TmpTransitionOne,t1/.default=activated,
1891 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
1892 t3/.store in=\TmpTransitionThree,t3/.default=activated,
1893 t4/.store in=\TmpTransitionFour,t4/.default=activated,
1894 t5/.store in=\TmpTransitionFive,t5/.default=activated,
1895 t6/.store in=\TmpTransitionSix,t6/.default=activated,
1896 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
1897 t8/.store in=\TmpTransitionEight,t8/.default=activated,
1898 t9/.store in=\TmpTransitionNine,t9/.default=activated,
1899 t10/.store in=\TmpTransitionTen,t10/.default=activated,
1900 scale/.store in=\TmpScale,scale/.default=0.5,
1901 x/.store in=\TmpX,x/.default=5,
1902 y/.store in=\TmpY,y/.default=5,
1903 }%
1904 }

1905 \tikzset{
1906   li petri/.style={
1907     activated/.style={
1908       very thick
1909     },
1910     inhibitor/.style={
1911       {Circle[open,length=2mm,fill=white]}-
1912     }
1913   }
1914 }

```

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

```

1915 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
1916 \def\liPetriTransitionsName#1{
1917   \ifmmode
1918     \liPetriTransitionsNameOhneMathe{#1}
1919   \else
1920     $\liPetriTransitionsNameOhneMathe{#1}$
1921   \fi
1922 }

```

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

```

1923 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
1924   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
1925 }

```

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

```

1926 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

1927 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm} }
1928

```

2.28 potenzmengen-konstruktion.sty

```
1929 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1930 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
1931 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
1932 \liLadePakete{formale-sprachen}
1933 \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}
    }
  }
}
```

```
1934 \def\liZustandsMengenSammlung#1#2{
1935   \liZustandsnameGross{#1}
1936   {
1937     \footnotesize
1938     \liPotenzmenge{
1939       \str_case:nn {#1} {#2
1940       }
1941     }
1942 }
```

```
\liZustandsMengenSammlungNr
```

```
1943 \def\liZustandsMengenSammlungNr#1#2{
1944   \liZustandsnameGross{#1}
1945   {
```

```
1946      \footnotesize
1947      \liZustandsmengeNr{
1948          \str_case:nn {#1} #2
1949      }
1950  }
1951 }

1952 \ExplSyntaxOff
1953
```

2.29 pseudo.sty

1954 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1955 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
 1956 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>Data: $G = (V, E, w)$: ein zusammenhängender, ungerichteter, kantengewichteter Graph kruskal(G)</p> <p>$E' \leftarrow \emptyset$; $L \leftarrow E$; Sortiere die Kanten in L aufsteigend nach ihrem Kantengewicht.;</p> <p>while $L \neq \emptyset$ do</p> <table border="0"> <tr> <td style="padding-left: 1em;">wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;</td> </tr> <tr> <td style="padding-left: 1em;">entferne die Kante e aus L;</td> </tr> <tr> <td style="padding-left: 1em;">if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then</td> </tr> <tr> <td style="padding-left: 2em;">$E' \leftarrow E' \cup \{e\}$;</td> </tr> <tr> <td style="padding-left: 1em;">end</td> </tr> </table> <p>end</p> <p>Result: $M = (V, E')$ ist ein minimaler Spannbaum von G.</p>	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 then	$E' \leftarrow E' \cup \{e\}$;	end
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 then					
$E' \leftarrow E' \cup \{e\}$;					
end					

1957 \RequirePackage[german,boxruled]{algorithm2e}

1958

2.30 pumping-lemma.sty

1959 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1960 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 1961 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 1962 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
1963 \def\liPumpingRegulaer{%
1964   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
1965   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
1966    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
1967   existiert, sodass die folgenden Eigenschaften erfüllt sind:
1968
1969   \begin{enumerate}
1970     \item  $|v| \geq 1$ 
1971     (Das Wort  $v$  ist nicht leer.)
1972
1973     \item  $|uv| \leq j$ 
1974     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
1975
1976     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
1977     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
1978     Sprache  $L$ )
1979   \end{enumerate}
1980
1981   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
1982   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
1983 }
```

\liPumpingKontextfrei

```
1984 \def\liPumpingKontextfrei{%
1985   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
1986   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
1987    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
1988
1989   \begin{enumerate}
1990     \item  $|vx| \geq 1$ 
1991     (Die Wörter  $v$  und  $x$  sind nicht leer.)
1992
1993     \item  $|vwx| \leq j$ 
1994     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
1995
1996     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^i w x^i y \in L$  (Für jede
1997     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w x^i y$  in der
1998     Sprache  $L$ )
1999   \end{enumerate}
2000 }
2001
```

2.31 quicksort.sty

```

2002 % https://tex.stackexchange.com/a/142634
2003 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2004 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2005
2006 %-----
2007 % USAGE:
2008 % \QSinitialize{comma, separated, numerical, values}
2009 % \loop
2010 % \QSpivotStep
2011 % \ifnum\value{pivotcount}>0
2012 %   \QSSortStep
2013 % \repeat
2014 %-----
2015
2016 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2017 \RequirePackage{xintfrac, xinttools}
2018
2019 \RequirePackage{tikz}
2020
2021 %-----
2022 % FIRST PART: TikZ styles and macros for the actual drawing
2023 \newcounter{cellcount}% used for coordinates of the node
2024 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2025
2026 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2027 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2028 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2029 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2030 % by police of LaTeX good conduct ? )
2031 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2032          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2033          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2034 % this is the "b" style as used in the image below
2035          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2036 % nicer:
2037          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2038          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2039
2040 % NOTE the b style was originally the same as the r(aised) style apart from
2041 % not being raised, but I find it nicer with a somewhat different
2042 % specification. I have not updated the images though.
2043
2044 % How the nodes are drawn depending on whether on the left of the pivot value
2045 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2046
2047 \def\DecoLEFT #1{%
2048   \xintFor* ##1 in {#1} \do
2049     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2050 }
2051
2052 \def\DecoINERT #1{%
2053   \xintFor* ##1 in {#1} \do
2054     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2055 }
2056
2057 \def\DecoRIGHT #1{%
2058   \xintFor* ##1 in {#1} \do
2059     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2060 }
2061
2062 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2063   \xintFor* ##1 in {#1} \do

```



```

2064     {\stepcounter{cellcount}}%
2065     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2066 }
2067
2068 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2069     \xintFor* ##1 in {#1} \do
2070     {\stepcounter{cellcount}}%
2071     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2072 }
2073
2074 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2075     \xintFor* ##1 in {#1} \do
2076     {\stepcounter{cellcount}}%
2077     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2078 }
2079
2080 %-----
2081 % SECOND PART: the actual sorting routines.
2082
2083 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2084 \def\QS@sort@b #1{\ifcase #1
2085     \expandafter\QS@sort@empty
2086     \or\expandafter\QS@sort@single
2087     \else\expandafter\QS@sort@c
2088     \fi
2089 }%
2090 \def\QS@sort@empty #1{}
2091 \def\QS@sort@single #1{\QS@Ir {#1}}
2092
2093 % This step is to pick the last as pivot.
2094 \def\QS@sort@c #1%
2095     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2096
2097 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2098 % The trick with \xintApplyUnbraced is that for example when selecting
2099 % the elements smaller than pivot, if we had been using \xintApply we
2100 % would have had at the minimum an empty brace pair. Thus we use the
2101 % "unbraced" variant, but then the \QS@select@smaller has added in
2102 % anticipation a level of braces.
2103 \def\QS@sort@d #1#2{%
2104     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2105     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2106     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2107 }%
2108 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2109 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2110 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2111
2112 %
2113 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2114 %
2115 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2116 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2117 % latter must handle correctly an empty argument.
2118
2119 %-----
2120 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2121
2122 % This draws all with suitable highlighting for the newly chosen pivots
2123 % (which will be shown raised)

```

```

2124 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2125             \let\QSIr\DecoINERT
2126             \let\QSIrr\DecoINERT
2127             \let\QSRr\DecoRIGHTwithPivot
2128 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2129             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2130             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2131 }
2132
2133 % This sorts and then draws, showing where the pivot chosen in the previous
2134 % step go. Next time they will have become "inert". If pivotcount is still at
2135 % zero on exit from \QSpivotStep, then this is the signal to stop before
2136 % executing \QSsortStep.
2137 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2138             \def\QSRr {\noexpand\QS@sort@a}%
2139             \def\QSIr {\noexpand\QSIrr}%
2140             \let\QSIrr\relax
2141             \edef\QS@list{\QS@list}%
2142             \let\QSLr\relax
2143             \let\QSRr\relax
2144             \let\QSIr\relax
2145             \edef\QS@list{\QS@list}%
2146             \let\QSLr\DecoLEFT
2147             \let\QSIr\DecoINERTwithPivot
2148             \let\QSIrr\DecoINERT
2149             \let\QSRr\DecoRIGHT
2150 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2151             \setcounter{cellcount}{0}%
2152             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2153 }
2154
2155 \def\QSinitialize #1{%
2156     % first, we convert the comma separated values into a list of braced items
2157     % we use an \edef, and anyhow many \edef's will be used later
2158     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2159     \let\QSRr\DecoRIGHT
2160     % The \QSRr marker mutated to draw the last element as
2161     % pivot and the earlier ones with the suitable style.
2162     %
2163     % The list of marked braced items \QS@list is used both for drawing
2164     % (as here) and for doing the exchange of elements during sort.
2165     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2166             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2167 }
2168

```

2.32 relationale-algebra.sty

```

2169 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2170 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2171 \RequirePackage{amsmath}
2172 \RequirePackage{amssymb}

```

```

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

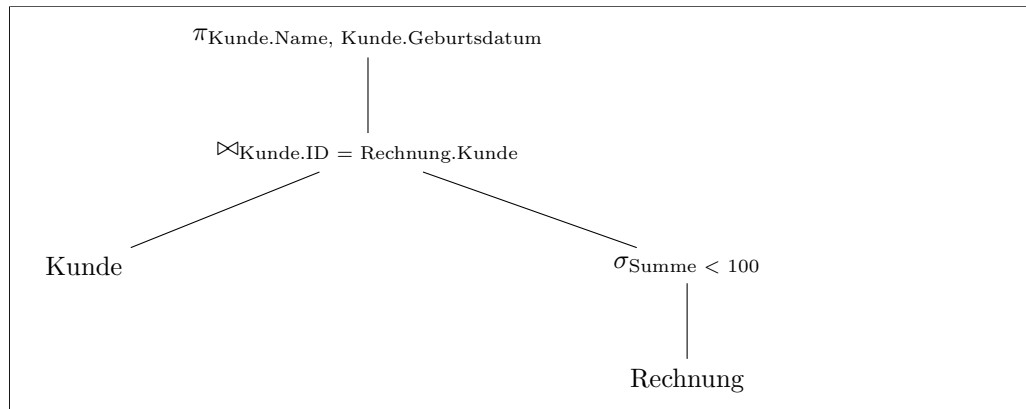
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2173 \RequirePackage{tikz}
2174 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2175 \def\o@join{\setbox0=\hbox{\bowtie}}%
2176   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2177 }

```

```
\leftouterjoin A \leftouterjoin B: A ⋈ B
```

```
2178 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

```
\rightouterjoin A \rightouterjoin B: A ⋉ B
```

```
2179 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

```
\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
```

```
2180 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

```
2181
```

2.33 rmodell.sty

```

2182 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2183 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2184 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2185 Datenbanken.]
2186 \RequirePackage{soul}

```

Let-Abkürzungen

```

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

```

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

```
2187 \def\liPrimaer#1{\ul{#1}}
```

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

```
2188 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```

2189 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2190 \ExplSyntaxOn
2191 \NewDocumentEnvironment { liRmodell }
2192 { +b }
2193 {
2194   \medskip
2195   {
2196     \linespread{2}
2197     \setlength{\parindent}{0pt}
2198     \li@Rmodell@Schrift#1
2199   }
2200   \medskip
2201 } {}
2202 \ExplSyntaxOff

```

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

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

```

2203 \def\liRelationMenge#1#2{
2204 \noindent
2205 #1 : \{ [ #2 ] \}
2206 \par
2207 }

```

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

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

```
2208 \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}

```

```
2209 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2210
```

2.34 sortieren.sty

```
2211 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2212 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2213 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} \nodepart{five} }
  \liSortierPfeil{one}{two}
  \liSortierPfeil{two}{three}
  \liSortierMarkierung{two split south}{three split north}
  \liSortierPfeilUnten{three}{one}
}
```



```
2214 \RequirePackage{tikz}
2215 \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.

```
2216 \def\liVertauschen#1{
2217   \directlua{
2218     local sortieren = require('lehramt-informatik-sortieren')
2219     sortieren('#1')
2220   }
2221 }
```

`\liSortierPfeil`

```
2222 \def\liSortierPfeil#1#2{
2223   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2224 }
```

`\liSortierPfeilUnten`

```
2225 \def\liSortierPfeilUnten#1#2{
2226   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2227 }
```

`\liSortierMarkierung`

```
2228 \def\liSortierMarkierung#1#2{\node[
2229   draw,
2230   very thick,
2231   fit=(reihe.#1) (reihe.#2),
2232   inner sep=0pt
2233 ] {}
2234 }

2235 \tikzset{
2236   li sortierung zahlenreihe/.style={
2237     draw,
2238     thin,
2239     font=\large,
2240     rectangle split horizontal,
2241     rectangle split,
2242   }
2243 }
```

```

2244 % https://tex.stackexchange.com/a/140895
2245 \RequirePackage{forest,xstring}
2246 \usetikzlibrary{calc}
2247
2248 \makeatletter
2249 \pgfmathdeclarefunction{strrepeat}{2}{%
2250   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2251   \let\pgfmathresult\pgfutil@empty
2252   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2253     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2254     \advance\pgfmath@count-1\relax
2255   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2256 \makeatother
2257
2258 \def\myNodes{}
2259
2260 \ExplSyntaxOn
2261 \newcommand*\sortList[1]{%
2262   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2263 \ExplSyntaxOff
2264
2265 \forestset{
2266   sort/.code={%
2267     \pgfmathparse{level()>\forestSortLevel}%
2268     \ifnum\pgfmathresult=0
2269       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2270       \sortList\myList
2271       \StrSubstitute{\myList}{,}{ }{\myList}%
2272       \pgfmathparse{strrepeat("1",level())}%
2273       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2274         (m\forestov{name}) {\myList}}%
2275       \pgfmathparse{level()=\forestSortLevel}%
2276       \ifnum\pgfmathresult=1
2277         \forestOget{\forestov{@first}}{name}\forestFirst
2278         \forestOget{\forestov{@last}}{name}\forestLast
2279         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2280           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2281       \fi
2282       \ifnum\forestov{@parent}=0\else
2283         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2284       \fi
2285       \gappto\myNodes{;}%
2286     \fi}}
2287
2288 \forestset{sort level/.code=%
2289   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2290   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2291

```

2.35 spalten.sty

```
2292 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2293 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2294 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2295 realisiert werden kann.]
2296 \RequirePackage{multicol}
```

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

```
2297 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2298
```

2.36 struktogramm.sty

```
2299 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2300 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2301 Paket struktex zum Setzen von Struktogrammen]
2302 \RequirePackage{struktex}
2303
```


2.37 syntax.sty

```
2304 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2305 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2306 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2307 \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
```

```
2308 \ExplSyntaxOn
2309 \directlua{
2310   syntax = require('lehramt-informatik-syntax')
2311   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2312   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2313   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2314   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2315   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2316   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2317 }
2318 \RequirePackage{hyperref}
2319 \RequirePackage{minted}
2320 % pygmentize -L styles
2321 \usemintedstyle{colorful}
2322 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2323 %\AfterEndEnvironment{minted}{\end{mdframed}}
2324 %\setminted{breaklines=true,linenos}
2325 \setminted{
2326   breaklines=true,
2327   linenos,
2328   fontsize=\footnotesize,
2329 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2330 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

`\liLatexCode` Im Zeilenfluss einen kurzen L^AT_EX-Code-Ausschnitt setzen.

```
2331 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2332 \def\li@GithubLink#1#2{
2333   \begin{flushright}
2334     \tiny
2335     Code-Beispiel~auf~Github~ansehen:~
2336     \href{#1}{\nolinkurl{#2}}
2337   \end{flushright}
2338 }
```

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

```
2339 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2340   \inputminted[#1]{java}{
2341     \directlua{
2342       syntax.drucke_absoluten_pfad('#2', false)
2343     }
2344   }
2345   \li@GithubLink
```

```

2346     {\directlua{syntax.drucke_github_url('#2', false)}}
2347     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2348 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2349 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2350   \inputminted[#1]{java}{
2351     \directlua{
2352       syntax.drucke_absoluten_pfad('#2', true)
2353     }
2354   }
2355   \li@GithubLink
2356   {\directlua{syntax.drucke_github_url('#2', true)}}
2357   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2358 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2359 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2360   \inputminted[#1]{java}{
2361     \directlua{
2362       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2363     }
2364   }
2365   \li@GithubLink
2366   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2367   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2368   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2369 }

\liAssemblerCode
2370 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2371 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2372   \inputminted{asm}{#1}
2373 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2374 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2375   \inputminted{componentpascal}{#1}
2376 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2377 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2378 \NewDocumentCommand{\liHaskellDatei}{ m }{
2379   \inputminted{haskell}{#1}
2380 }

2381 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2382 \def\liSqlCode#1{\mintinline{sql}|#1|}
2383

```

2.38 syntaxbaum.sty

```
2384 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2385 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2386 Syntaxbäumen mit Hilfe des Pakets tikz-qtree]
2387 \RequirePackage{tikz-qtree}
2388
2389 \tikzset{li parsetree/.style={
2390     every internal node/.style={
2391         draw,circle
2392     },
2393     every leaf node/.style={
2394         draw,rectangle
2395     },
2396 }
2397 }
2398
```

2.39 synthese-algorithmus.sty

```

2399 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2400 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2401 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2402 Relation in die 3. Normalform]

2403 \liLadePakete{normalformen,mathe,typographie}
2404 \ExplSyntaxOn

\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}_α 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 \mathcal{R}_α , die in einem anderen Relationenschema $\mathcal{R}_{\alpha'}$ enthalten sind, d. h. $\mathcal{R}_\alpha \subseteq \mathcal{R}_{\alpha'}$. —

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

```

2405 \def\liSyntheseUeberschrift#1{
2406   {
2407     \bfseries
2408     \sffamily
2409     \str_case:nn {#1} {
2410       {1} {Kanonische-Überdeckung}
2411       {1-1} {Linksreduktion}
2412       {1-2} {Rechtsreduktion}
2413       {1-3} {Löschen-leerer-Klauseln}
2414       {1-4} {Vereinigung}
2415       {2} {Relationsschemata-formen}
2416       {3} {Schlüssel-hinzufügen}
2417       {4} {Entfernung-überflüssiger-Teilschemata}
2418     }
2419   }
2420 }

```

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

```

2421 \def\li@synthese@erklaerung@texte#1{
2422   \str_case:nn {#1} {
2423     {1} {
2424       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2425       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2426       Schritten-erreicht-werden.
2427     }
2428     {1-1} {
2429       Führe-für-jede-funktionale-Anhängigkeit-
2430       $\alpha\rightarrow\beta$-in-$F$-die-Linksreduktion-durch,-
2431       überprüfe-also-für-alle-
2432       $A\in\alpha$,~ob-$A$-überflüssig-ist,-d.h.-ob-
2433       $\beta\subseteqq\liAttributHuelle{F,\alpha\cup A}$.
2434     }
2435     {1-2} {
2436       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2437       alle-$B\in\beta$,~ob-$B\in\liAttributHuelle{F\cup(\alpha\rightarrow\beta)}$,~
2438       $\alpha\rightarrow\beta\cup(\alpha\rightarrow\beta)$,~
2439       $\alpha$-gilt.-In-diesem-Fall-ist-$B$-auf-der-rechten-Seite-
2440       überflüssig-und-kann-eliminiert-werden,-dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow(\beta\cup B)$-
2441       ersetzt.
2442     }
2443     {1-3} {
2444       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2445       entstanden-sind.
2446     }
2447     {1-4} {
2448       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2449       der-Form-$\alpha\rightarrow\beta\sb{1},\dots,\alpha\rightarrow\beta\sb{n}$,-so-dass-$\alpha\rightarrow\beta\sb{1}\cup\dots\cup\beta\sb{n}$-verbleibt.
2450     }
2451     {2} {
2452       % Kemper Seite 197
2453       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta\sb{c}$-ein-Relationenschema-$\mathcal{R}\sb{\alpha}$-
2454       :=$\alpha\cup\beta$.
2455     }
2456     {3} {
2457       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata-$R\sb{\alpha}$-
2458       einen-Schlüsselkandidaten-von-$\mathcal{R}$-bezüglich-$F\sb{c}$-

```

```

2465 enthält,~sind-wir~fertig,~sonst~wähle-einen-Schlüsselkandidaten-
2466  $\mathcal{K} \sim \text{subseq} \mathcal{R}$ ~aus-und-definiere~folgendes-
2467 zusätzliche-Schema:~ $\mathcal{R} \setminus \mathcal{K} \sim \mathcal{K}$ ~und- $\mathcal{F} \setminus \mathcal{K} \sim \emptyset$ 
2468 }
2469 {4} {
2470 Eliminiere~diejenigen-Schemata- $\mathcal{R} \setminus \alpha$ ~,~die~in~einem~
2471 anderen-Relationenschema- $\mathcal{R} \setminus \alpha'$ ~enthalten~sind,~d.h.~
2472  $\mathcal{R} \setminus \alpha \sim \text{subseq} \mathcal{R} \setminus \alpha'$ ~.
2473 }
2474 }
2475 }
2476 }
2477 \def\liSyntheseErklaerung#1{
2478 {
2479 \itshape
2480 \footnotesize
2481 \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2482 }
2483 }

```

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

```

2484 \def\liSyntheseUeberErklaerung#1{
2485 \liSyntheseUeberschrift{#1}\par
2486 \liSyntheseErklaerung{#1}
2487 }

```

```

2488 \ExplSyntaxOff
2489

```

2.40 tabelle.sty

2490 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2491 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2492 \RequirePackage{tabularx}

2493

2.41 typographie.sty

```
2494 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2495 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2496 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2497 formatierung.sty definiert.]
```

```
2498 \ExplSyntaxOn
```

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

```
2499 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2500 \let\liErledigt=\faCheckSquareO
```

```
\liNichtsZuTun \liNichtsZuTun: ☑ Nichts zu tun
```

```
2501 \def\liNichtsZuTun{\faCheckSquareO{}}~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 —

```
2502 \def\liParagraphMitLinien#1{
```

```
2503 \noindent
```

```
2504 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2505 \enspace
```

```
2506 #1
```

```
2507 \enspace
```

```
2508 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2509 \par
```

```
2510 \medskip
```

```
2511 }
```

```
2512 \ExplSyntaxOff
```

```
2513
```


2.42 uml.sty

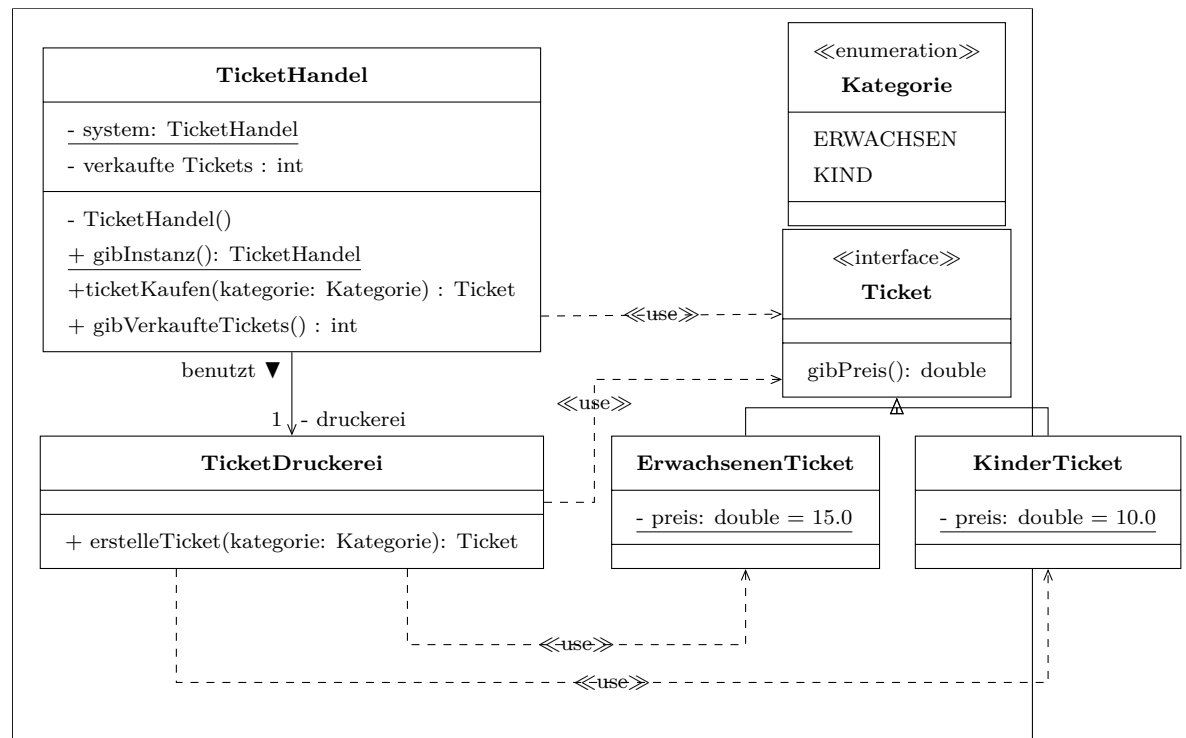
```

2514 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2515 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2516 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2517 Erweiterung bereitstellt]

2518 \RequirePackage{tikz-uml}
2519 \RequirePackage{tikz-uml-activity}
2520 % Not compatible with wasysym
2521 %\RequirePackage{mathabx}
2522 \RequirePackage{wasysym}
2523 \usetikzlibrary{positioning}

2524 \tikzumlset{
2525   fill class=white!0,
2526   font=\footnotesize,
2527   fill object=white!0,
2528   fill note=white!0,
2529   fill state=white!0,
2530   % Use case
2531   fill usecase=white!0,
2532   fill system=white!0,
2533 }

```



```

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

```

```

2534 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2535   \def\@liDirLeft{
2536     \def\@liDirRight{ \RIGHTarrow}
2537     \pgfkeys{/lese/dir/.is choice}
2538     \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2539     \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2540     \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{\def\@liDirLeft{\LEFTarrow }}}
2541     \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2542
2543     \def\@liPos{above}
2544     \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2545

```

```

2546 \def\@liDistance{0cm}
2547 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2548
2549 \pgfkeys{/lese/.cd,#1}
2550
2551 \node[\@liPos = \@liDistance of #2-middle] {
2552   \@liDirLeft{\footnotesize#2}\@liDirRight
2553 };
2554 }
2555

```

2.43 vollstaendige-induktion.sty

2556 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2557 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2558 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2559 Überschriften für die einzelnen Schritte]

Let-Abkürzungen

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

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

Lade häufig benötigte Pakete

2560 \RequirePackage{lehramt-informatik-typographie}
 2561 \RequirePackage{lehramt-informatik-mathe}
 2562 \RequirePackage{lehramt-informatik-syntax}
 2563 \ExplSyntaxOn

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

2564 \def\liInduktionMarkierung#1{\textcolor{blue}{#1}}

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

Let-Abkürzung: \let\e=\liInduktionErklaerung

```
2565 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2566 \def\liInduktionAnfang{
```

```
2567   \liPseudoUeberschrift{Induktionsanfang}
```

```
2568
```

```
2569   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2570   \liParagraphMitLinien{
```

```
2571     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
2572   }
```

```
2573 }
```

\liInduktionVoraussetzung

```
2574 \def\liInduktionVoraussetzung{
```

```
2575   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
2576
```

```
2577   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2578   \liParagraphMitLinien{
```

```
2579     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ .
```

```
2580   }
```

```
2581 }
```

\liInduktionSchritt

```
2582 \def\liInduktionSchritt{
```

```
2583   \liPseudoUeberschrift{Induktionsschritt}
```

```
2584
```

```
2585   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2586   \liParagraphMitLinien{
```

```
2587     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
2588     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
2589   }
```

```
2590 }
```

```
2591 \ExplSyntaxOff
```

```
2592
```

2.44 wasserfall.sty

```
2593 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2594 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2595 \RequirePackage{tikz}
2596 \tikzset{wasserfall/.style={
2597   >=stealth,
2598   node distance = 2mm and -8mm,
2599   start chain = A going below right,
2600   every node/.style = {
2601     draw,
2602     text width=24mm,
2603     minimum height=12mm,
2604     align=center,
2605     inner sep=1mm,
2606     fill=white,
2607     drop shadow={fill=black},
2608     on chain=A
2609   },
2610 }}
2611 \usetikzlibrary{chains,positioning,shadows}
2612
```

2.45 wpkalkuel.sty

2613 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2614 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2615 \RequirePackage{amsmath}

2616 \ExplSyntaxOn

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

2617 \def\liWpKalkuelOhneMathe#1#2{

2618 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2619 }

2620 \def\liWpKalkuel#1#2{

2621 \ifmmode

2622 \liWpKalkuelOhneMathe{#1}{#2}

2623 \else

2624 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2625 \fi

2626 }

\MatheEnv

2627 \def\MatheEnv#1{

2628 \medskip

2629

2630 \hspace{1em}#1

2631

2632 \medskip

2633 }

\Mathe

2634 \def\Mathe#1{

2635 \MatheEnv{#1\$}

2636 }

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

2637 \def\liWpEquivalent#1{

2638 \MatheEnv{\$\equiv\$\hspace{1em}#1\$}

2639 }

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

2640 \newlength{\@Skip@Erklaerung@Reset}

2641 \def\liWpErklaerung#1{

2642 \setlength{\@Skip@Erklaerung@Reset}{\leftskip}

2643 \setlength{\leftskip}{0.5cm}

2644

2645 \par

2646 \noindent

2647 {

2648 \scriptsize

2649 #1

2650 }

2651 \par

2652

2653 \setlength{\leftskip}{\@Skip@Erklaerung@Reset}

2654 }

```

2655 \def\liWpErklaerungVerzweigung{
2656   $\liWpKalkuelOhneMathe{if~\{~b~\}~then~\{~a1~\}~else~\{~a2~\}\}{Q}
2657   \equiv
2658   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2659   \lor
2660   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2661 }

2662 \ExplSyntaxOff

2663

```

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	\alpha 2430, 2432, 2433,	\bowtie
\# 109	2436, 2438, 2439,	2175, 2178, 2179, 2180
\, 331, 386,	2440, 2441, 2442,	\Box 159
1063, 1065, 1551,	2446, 2452, 2453,	\boxtimes 463
1552, 1553, 1860, 2330	2458, 2459, 2460,	
\@Skip@Erklaerung@Reset	2463, 2471, 2472, 2473	C
... 2640, 2642, 2653	\arabic 1126, 2049, 2054,	\c 1209, 1210
\@afterheading 1518	2059, 2065, 2071, 2077	\cdot 1595, 1648, 1659
\@afterindentfalse . 1518	\arraystretch 1777	\centerline
\@liDirLeft 2535, 2540, 2552		1240, 2128, 2150, 2165
\@liDirRight 2536, 2538,	B	\chapter 1116, 1117
2539, 2540, 2541, 2552	\BeforeBeginEnvironment	\char 1425
\@liDistance 2322	\clearpage 1419
... 2546, 2547, 2551	\begin .. 614, 656, 671,	\cline 596
\@liPos .. 2543, 2544, 2551	695, 719, 753, 772,	\clist 226, 270,
\ 596, 616,	782, 790, 810, 838,	271, 284, 288, 2262
617, 620, 621, 624,	855, 881, 902, 919,	\columnbreak 2297
625, 697, 698, 699,	934, 1042, 1046,	\cs 287, 306, 330,
774, 776, 794, 803,	1132, 1207, 1235,	331, 368, 380, 1501
844, 883, 884, 885,	1244, 1251, 1359,	\csname 1194, 1197
890, 891, 892, 908,	1417, 1469, 1474,	\cup 1021, 2439, 2453, 2460
1425, 1782, 1839, 1842	1482, 1507, 1512,	
\{ 207, 998, 1008,	1523, 1533, 1537,	D
1020, 1021, 1026,	1620, 1624, 1640,	\DeclareMathSymbol ..
1040, 1062, 1281,	1663, 1686, 1701, 1725, 1726
1827, 1850, 2205, 2656	1780, 1781, 1831,	\DecoINERT
\} 207, 998, 1008,	1969, 1989, 2130,	2052, 2125, 2126, 2148
1020, 1021, 1028,	2152, 2166, 2322, 2333	\DecoINERTwithPivot .
1046, 1066, 1282,	\begingroup 1418, 1829, 2250 2068, 2147
1827, 1854, 2205, 2656	\beschriftung	\DecoLEFT 2047, 2146
_ 27, 35, 44, 46, 287, 306, 1449, 1453,	\DecoLEFTwithPivot ..
330, 331, 345, 346,	1457, 1461, 1465, 1467 2062, 2124
352, 355, 358, 368, 380	\beta 2430,	\DecoRIGHT 2057, 2149, 2159
	2433, 2437, 2438,	\DecoRIGHTwithPivot .
A	2439, 2442, 2452, 2074, 2127
\addbibresource	2453, 2454, 2459, 2460	\definecolor 1114
1395, 1396, 1397,	\bf 2027, 2028, 2029	\delta 65, 107, 165, 207, 1012
1398, 1399, 1400,	\bfseries .. 475, 1116,	\dh 1553, 2441
1401, 1402, 1403, 1404	1118, 2027, 2033,	\directlua
\advance 2254	2035, 2037, 2038, 2407	58, 137, 195, 200,
\AfterEndEnvironment 2323	\bigskip 48, 364,	992, 1007, 1027,
\allsectionsfont ... 1112	598, 603, 1427, 1755	1035, 1043, 1049,
\Alpha 1126	\bool 309, 332	1812, 1817, 1845,
\alph 1126, 1127		1852, 1857, 2217,

2309, 2341, 2346, 2347, 2351, 2356, 2357, 2361, 2367, 2368	\erzeuge@tiefgestellt ... 1007, 1008, 1012	\fullouterjoin 2180
\do 2048, 2053, 2058, 2063, 2069, 2075	\expandafter 1194, 2083, 2085, 2086, 2087, 2095, 2253	G
\dots 506, 510, 1281, 1976, 2452, 2453	\ExplSyntaxOff . 50, 92, 134, 139, 192, 197, 202, 393, 528, 550, 565, 986, 1068, 1103, 1379, 1543, 1618, 1805, 1952, 2202, 2263, 2381, 2488, 2512, 2591, 2662	\g 29, 37, 270, 271, 284, 288, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 307, 311, 312, 313, 316, 318, 319, 320, 321, 322, 323, 334, 335, 336, 337, 345, 346, 348, 354, 355, 357, 358, 360, 361, 369, 374, 376, 381, 383, 387
\DOWNarrow 2539	\ExplSyntaxOn .. 22, 61, 102, 135, 160, 193, 198, 223, 470, 534, 551, 959, 1058, 1080, 1372, 1444, 1557, 1787, 1933, 2190, 2260, 2308, 2404, 2498, 2563, 2616	\Gamma . 106, 164, 207, 1021
\draw 1214, 1217, 1220, 1924, 2223, 2226		\gappto 2285
E		\geq 1288, 1965, 1970, 1986, 1990
\edef 1205, 2141, 2145, 2157, 2158		H
\else 570, 578, 586, 1002, 1016, 1054, 1465, 1568, 1578, 1588, 1602, 1823, 1919, 2087, 2280, 2282, 2623		\hbox 2175
\emph 962, 1262, 1291, 1293, 1424		\headrulewidth 1142
\empty 1465	F	\hfill 1644, 1650, 1655, 2508
\emptyset 1769, 2447, 2468	\faCheckSquare0 2500, 2501	\hinweis 1440
\end ... 653, 667, 692, 716, 750, 769, 779, 787, 806, 835, 852, 877, 899, 915, 931, 949, 1044, 1046, 1134, 1223, 1247, 1248, 1255, 1364, 1422, 1471, 1479, 1498, 1514, 1515, 1529, 1540, 1541, 1622, 1635, 1660, 1683, 1698, 1708, 1784, 1785, 1833, 1979, 1999, 2130, 2152, 2166, 2323, 2337	\faCircleThin 978	\hline 1782
\endcsname 1194, 1197	\faGg 972	\href 370, 1076, 1549, 1712, 2336
\endgroup 1421, 1834, 2255	\fancyfoot 1138, 1139, 1140	\hspace .. 1927, 2630, 2638
\enspace 2505, 2507	\fancyhead 1137	\ht 2176
environments:	\faSquare0 966	I
liAdditum 1472	\fi 572, 580, 588, 1004, 1018, 1056, 1468, 1570, 1580, 1590, 1604, 1825, 1921, 2088, 2280, 2281, 2284, 2286, 2625	\i 1219, 1220
liAHuelle 1828	\fontspec 1112	\ifcase 2084
liAntwort 1443	\footcite 668, 690, 726, 749, 768, 807, 834, 876, 1260, 1263, 1270, 1275, 1280, 1284, 1290, 1295, 1384, 1636, 1637, 1792, 1982	\ifmmode 568, 576, 584, 1000, 1014, 1052, 1566, 1576, 1586, 1600, 1821, 1917, 2621
liDiagramm 1531	\footnote 1545, 1549	\IfNoValueTF 1475, 1545, 1549
liEinbettung 1442	\footnotesize 147, 341, 424, 520, 983, 1440, 1493, 1508, 1524, 1736, 1830, 1937, 1946, 2189, 2328, 2480, 2526, 2552	\ifnum 2011, 2252, 2268, 2276, 2282
liExkurs 1480	\foreach . 1209, 1212, 1219	\ifx 1465, 2280
liGraphenFormat . 1186	\forestFirst .. 2277, 2280	\in 494, 600, 1288, 1610, 1613, 1616, 1642, 1648, 1653, 1965, 1976, 1986, 1996, 2430, 2432, 2438, 2459, 2579
liKasten 1131	\forestLast ... 2278, 2280	\inhaltsverzeichnis 1416
liKontrollflussgraph 1358	\forestOget ... 2277, 2278	\input . 4, 7, 10, 13, 16, 394
liLernkartei 1521	\forestOnes 2290	\inputminted 2340, 2350, 2360, 2372, 2375, 2379
liProduktionsRegeln 1036	\forestOv 2279, 2280, 2283	\int 2262
liProjektSprache 1441	\forestov . 2269, 2273, 2274, 2277, 2278, 2279, 2280, 2282, 2283	\item 463, 464, 673, 677, 682, 687, 720, 729, 734, 742, 783, 811, 816, 823, 831, 856, 861, 865, 870, 935, 940, 945, 1245, 1246, 1501, 1505, 1625, 1628, 1632, 1641, 1647, 1652, 1664,
liQuellen 1501	\forestset 2265, 2288	
liRelationenSchemaFormat 2209	\forestSortLevel 2267, 2275, 2289, 2290	
liRmodell 2189	\frac 1597, 1628, 1659, 1674	
liUebergangsTabelle 1777		
\equiv 2638, 2657		

1668, 1672, 1676, 1680, 1687, 1691, 1695, 1970, 1973, 1976, 1990, 1993, 1996	\itshape 519, 2479	\LehramtInformatikGithubTexRepoBedingungEins 372, 2314	\LehramtInformatikGithubTexRepoBedingungEins 1609, 1644, 1687
J		\LehramtInformatikRepository\liBedingungFalsch 4,	\liBedingungFalsch 1369
\j 1209, 1210, 1212, 1213, 1214, 1219, 1220, 1221		7, 10, 13, 16, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403, 1404, 2311	\liBedingungWahr 1368 \liBedingungZwei 1612, 1650, 1691 \liBeschriftung 1432 \liChomskyErklaerung 485, 526
K		\leq 1659, 1973, 1993	\liChomskyUeberErklaerung 524
\k 1219		\let 1010, 1011, 1419, 2124, 2125, 2126, 2127, 2140, 2142, 2143, 2144, 2146, 2147, 2148, 2149, 2159, 2251, 2289, 2290, 2500	\liChomskyUeberschrift 473, 525 \liCpmEreignis 534 \liCpmFruehesterI 591 \liCpmSpaetesterI 590
\keys 31, 70, 82, 112, 122, 170, 180, 292, 538, 542, 556, 561, 1087, 1094		\li@chomsky@erklaerung@texte 485, 521	\liCpmVon 574 \liCpmVonOhneMathe 574, 577, 579
L		\li@fussnote@text 1733, 1739, 1743, 1747, 1751	\liCpmVonZu 566 \liCpmVonZuOhneMathe 566, 569, 571
\l 63, 64, 65, 66, 67, 68, 71, 72, 73, 74, 75, 77, 79, 84, 85, 86, 87, 88, 89, 104, 105, 106, 107, 108, 109, 110, 113, 114, 115, 116, 117, 118, 119, 125, 126, 127, 128, 129, 130, 131, 162, 163, 164, 165, 166, 167, 168, 171, 172, 173, 174, 175, 176, 177, 183, 184, 185, 186, 187, 188, 189, 536, 539, 544, 545, 548, 553, 554, 557, 558, 563, 1082, 1083, 1084, 1085, 1088, 1089, 1090, 1091, 1097, 1098, 1099, 1100, 1375, 1376, 1377, 1504, 1505, 1506, 1513		\li@GithubLink 2332, 2345, 2355, 2366 \li@mget 1196, 1200, 1220 \li@minc 1199, 1221 \li@mset 1193, 1201, 1210, 1213 \li@numdiscs 1205, 1214, 1220 \li@Rmodell@Schrift 2189, 2198, 2208 \li@sequence 1206, 1219 \li@synthese@erklaerung@texte 2421, 2481	\liCpmVorgang 551 \liCpmZu 582 \liCpmZuOhneMathe 582, 585, 587 liDiagramm (environ- ment) 1531 liEinbettung (environ- ment) 1442 \liEntwurfsAbstrakteFabrikUml 613 \liEntwurfsAdapterAkteure 670 \liEntwurfsAdapterUml 655 \liEntwurfsBeobachterAkteure 718 \liEntwurfsBeobachterUml 694 \liEntwurfsDekoriererUml 752 \liEntwurfsEinzelstueckAkteure 781 \liEntwurfsEinzelstueckUml 771 \liEntwurfsErbauerAkteure 809 \liEntwurfsErbauerUml 789 \liEntwurfsFabrikmethodeAkteure 854 \liEntwurfsFabrikmethodeUml 837 \liEntwurfsKompositumUml 880 \liEntwurfsModellPraesentationSteuerungUml 901 \liEntwurfsZustandAkteure 933 \liEntwurfsZustandUml 918 \liEpsilon 1006 \liErAttribute 962, 976, 978 \liErDatenbankName 981
\labelenumi 1127		\li@numdiscs 1205, 1214, 1220	
\labelenumii 1128		\li@Rmodell@Schrift 2189, 2198, 2208	
\labelitemi 1121		\li@sequence 1206, 1219	
\labelitemii 1122		\li@synthese@erklaerung@texte 2421, 2481	
\labelitemiii 1123		\li@Ableitung 1035	
\labelitemiv 1124		liAdditum (environment) 1472	
\land 2658, 2660		liAHuelle (environment) 1828	
\LARGE 1116		\liAlphabet 1020	
\large 1240, 2239		liAntwort (environment) 1443	
\leaders 2508		\liAnweisung 1366	
\left 1560		\liAssemblerCode 2370	
\LEFTarrow 2540		\liAssemblerDatei 2371	
\leftarrow 582		\liAttribut 2208	
\leftouterjoin 2178		\liAttributHuelle 1819, 2433, 2438	
\leftskip 2642, 2643, 2653		\liAttributHuelleLinksReduktion 1836	
\LehramtInformatikGitBranch 373, 2316		\liAttributHuelleOhneMathe 1819, 1822, 1824, 1838	
\LehramtInformatikGithubCodeRepo 2315		\liAttributMenge 1827, 1838, 1841	
\LehramtInformatikGithubDomain 2312		\liAufgabe 3	
\LehramtInformatikGithubRawDownload 371, 2313		\liAufgabenTitel 23	
		\liAusdruck 1059	
		\liAutomat 61	
		\liAutomatenKante 93	
		\liBandAlphabet 1021	
		\liBedingung 1367	
		\liBedingungDrei 1615, 1655, 1695	

<code>\liErEntity</code> . 960, 964, 966	<code>\liLadePakete</code> 54, 57, 224, 229, 472, 533, 1230, 1731, 1811, 1932, 2403	<code>\liProblemSubsetSum</code> 1277, 1286
<code>\liErledigt</code> 2500	<code>\liLatexCode</code> 2331	<code>\liProblemVertexCover</code> 1257, 1265
<code>\liErMpAttribute</code> 975	<code>\liLeereZelle</code> 1769	<code>\liProduktionen</code> 1048, 1090
<code>\liErMpEntity</code> 963	<code>liLernkartei</code> (environ- ment) 1521	<code>liProduktionsRegeln</code> (environment) 1036
<code>\liErMpRelationship</code> . 969	<code>\liMasterExkurs</code> 1700	<code>liProjektSprache</code> (envi- ronment) 1441
<code>\liErRelationship</code> 961, 970, 972	<code>\liMasterFaelle</code> 1639, 1707	<code>\liPseudoUeberschrift</code> 1426, 1476, 1477, 1779, 1789, 2567, 2575, 2583
<code>\liExamensAufgabe</code> 6	<code>\liMasterFallRechnung</code> 1685	<code>\liPumpingKontextfrei</code> 1984
<code>\liExamensAufgabeA</code> . . . 15	<code>\liMasterVariablen</code> 1619, 1702	<code>\liPumpingRegulaer</code> . 1963
<code>\liExamensAufgabeTA</code> . . 12	<code>\liMasterVariablenDeklaration</code> 1662	<code>liQuellen</code> (environment) 1501
<code>\liExamensAufgabeTTA</code> . . 9	<code>\liMasterWolframLink</code> 1710	<code>\liRekursionsGleichung</code> 1606, 1666
<code>liExkurs</code> (environment) 1480	<code>liMenge</code> . . . 71, 72, 74, 113, 114, 115, 119, 171, 172, 173, 177, 998, 1049, 1088, 1089	<code>\liRelation</code> 1856
<code>\liFalsch</code> 464	<code>\liMengeOhneMathe</code> 998, 1001, 1003	<code>liRelationenSchemaFormat</code> (environment) 2209
<code>\liFlaci</code> 1069	<code>\liMinimierungErklaerung</code> 1791	<code>\liRelationMenge</code> . . . 2203
<code>\liFremd</code> 2188	<code>\liMinispracheDatei</code> 2374	<code>\liRichtig</code> 463
<code>\liFunktionaleAbhaengigkeit</code> 1844	<code>\linespread</code> 2196	<code>liRmodell</code> (environment) 2189
<code>\liFunktionaleAbhaengigkeith</code> 1847	<code>\liNichtsZuTun</code> 2501	<code>\liRundeKlammer</code> . 1559, 1563, 1573, 1583, 1597
<code>\liFussnote</code> . . . 1732, 1734	<code>\liO</code> 1582, 1610	<code>\liSetzeAufgabenTitel</code> . 25
<code>\liFussnoteDreiText</code> 1746, 1764	<code>\liOmega</code> 1572, 1616	<code>\liSortierMarkierung</code> 2228
<code>\liFussnoteEinsText</code> 1738, 1758	<code>\liOmegaOhneMathe</code> 1572, 1577, 1579	<code>\liSortierPfeil</code> 2222
<code>\liFussnoteLink</code> 1548	<code>\liOOhneMathe</code> 1582, 1587, 1589	<code>\liSortierPfeilUnten</code> 2225
<code>\liFussnoten</code> 1754	<code>\liParagraphMitLinien</code> 521, 1793, 2481, 2502, 2570, 2578, 2586	<code>\liSpaltenUmbruch</code> . . 2297
<code>\liFussnoteUrl</code> . 916, 1544	<code>\liPetriErreichKnotenDrei</code> 1926	<code>\liSqlCode</code> 2382
<code>\liFussnoteVierText</code> 1750, 1767	<code>\liPetriErreichTransition</code> 1923	<code>\listen@punkt</code> . . 1501, 1513
<code>\liFussnoteZweiText</code> 1742, 1761	<code>\liPetriSetzeSchluessel</code> 1868	<code>\liStrich</code> 1232
<code>\liGrammatik</code> 1080	<code>\liPetriTransitionsName</code> 1915, 1927	<code>\liSyntheseErklaerung</code> 2421, 2486
<code>liGraphenFormat</code> (envi- ronment) 1186	<code>\liPetriTransitionsNameOhneMathe</code> 1915, 1918, 1920	<code>\liSyntheseUeberErklaerung</code> 2484
<code>\liHanoi</code> 1193	<code>\liPetriTransPfeile</code> 1927	<code>\liSyntheseUeberschrift</code> 2405, 2485
<code>\liHaskellCode</code> 2377	<code>\liPolynomiellReduzierbar</code> 1250	<code>\liT</code> 1592, 1607, 1621, 1682
<code>\liHaskellDatei</code> 2378	<code>\liPotenzmenge</code> 1007, 1011, 1938	<code>\liTeilen</code> 1816
<code>\liInduktionAnfang</code> . 2566	<code>\liPotenzmengeOhneMathe</code> 1008, 1009, 1010	<code>\liLiteratur</code> 1383, 1407
<code>\liInduktionErklaerung</code> 2565	<code>\liPrimaer</code> 2187	<code>\liTheta</code> 1562, 1613, 1642, 1648, 1653
<code>\liInduktionMarkierung</code> 2564	<code>\liProblemBeschreibung</code> 1234	<code>\liThetaOhneMathe</code> 1562, 1567, 1569
<code>\liInduktionSchritt</code> 2582	<code>\liProblemClique</code> . . . 1257	<code>\liTOhneMathe</code> 1592, 1601, 1603
<code>\liInduktionVoraussetzung</code> 2574	<code>\liProblemName</code> 1233, 1240, 1252, 1254, 1267, 1278, 1279, 1287, 1288	<code>\liTuringKante</code> 203
<code>\liJavaCode</code> 2330	<code>\liProblemSat</code> 1286	<code>\liTuringLeerzeichen</code> 159, 167
<code>\liJavaDatei</code> 2339		<code>\liTuringMaschine</code> . . . 160
<code>\liJavaExamen</code> 2359		<code>\liTuringUeberfuehrung</code> 206
<code>\liJavaTestDatei</code> . . . 2349		<code>\liTuringUebergaenge</code> 198, 204
<code>liKasten</code> (environment) 1131		<code>\liTuringUebergangZelle</code> 193
<code>\liKellerAutomat</code> 102		
<code>\liKellerKante</code> 140		
<code>\liKellerUebergang</code> 135, 141		
<code>\liKontrollCode</code> 1370		
<code>liKontrollflussgraph</code> (environment) 1358		
<code>\liKontrollKnotenPfad</code> 1372		
<code>\liKontrollTextzeileKnoten</code> 1371, 1376		
<code>\liKurzeTabellenLinie</code> 596		
<code>\liLadeAllePakete</code> . . . 228		

Q		2319, 2387, 2492,	\stepcounter 2049, 2054,
\QS@list	2130, 2141, 2145,	2499, 2518, 2519,	2059, 2062, 2064,
	2152, 2158, 2163, 2166	2521, 2522, 2560,	2068, 2070, 2074, 2076
\QS@select@equal 2105, 2109	2561, 2562, 2595, 2615	\str .. 477, 486, 1447,
\QS@select@greater 2106, 2110	\right 1560	1939, 1948, 2409, 2422
\QS@select@smaller 2101, 2104, 2108	\RIGHTarrow ... 2536, 2541	\StrSubstitute . 2269, 2271
\QS@sort@a	2083, 2116, 2137, 2138	\Rightarrow 600, 605	\strut 1664, 1668,
\QS@sort@b	2083, 2084	\rightarrow 207,	1672, 1676, 1680, 2297
\QS@sort@c	2087, 2094	489, 494, 502, 506,	\subseteq 2433, 2466, 2473
\QS@sort@d	2095, 2103	508, 509, 511, 566,	
\QS@sort@empty .	2085, 2090	574, 1927, 2430,	T
\QS@sort@single 2086, 2091		2437, 2439, 2442,	\tableofcontents ... 1420
\QSinitialize 2008, 2120, 2155	2447, 2452, 2453, 2458	\text 77, 79,
\QSIr . 2091, 2097, 2105,	2125, 2139, 2144, 2147	\rightouterjoin 2179	182, 1819, 2565, 2618
\QSIrr 2126, 2139, 2140, 2148		\Roman 1126	\textbf 960, 1258,
\QSLr	2097,	\roman 1126, 1128	1267, 1278, 1287,
	2104, 2115, 2116,	\romannumeral 2095	1429, 1436, 1467,
	2124, 2137, 2142, 2146	\rtimes 1718	1495, 1510, 1526, 1782
\QSpivotStep	2010, 2120, 2124, 2135	\rule 2128, 2150, 2165, 2176	\textcolor 1370, 2564
\QSR	2097		\textit
\QSRr	2106,	S	844, 883, 884, 885,
	2127, 2138, 2143,	\sb 67, 77,	886, 1536, 1827, 1860
	2149, 2158, 2159, 2160	79, 108, 166, 501,	\textsc 1233
\QSSortStep	2012, 2120, 2136, 2137	502, 506, 509, 510,	\textsf 1429, 1510
\quad	1735	511, 1053, 1055,	\textstyle 1628, 1659
		1610, 1613, 1616,	\texttt 983, 1233, 1367,
R		1642, 1648, 1797,	1368, 1369, 1370, 2618
\raisebox	1371	1915, 1924, 2452,	\thepage 1140
\relax	1419,	2453, 2454, 2459,	\theparagraph 1118
	2097, 2140, 2142,	2463, 2464, 2467,	\Theta 1563
	2143, 2144, 2252, 2254	2468, 2471, 2472, 2473	\thinspace 2618
\renewcommand 1121, 1122,	\scriptscriptstyle ..	\tikz 1371
	1123, 1124, 1127, 566, 574, 582	tikz: bbaum 23
	1128, 1142, 1143, 1777	\scriptsize	tikz: li binaer baum 21
\repeat	2013	1072, 1311, 1318,	\tikzchildnode 419
\RequirePackage 55, 158,	222, 226, 395, 401,	1324, 2565, 2618, 2648	\tikzparentnode 419
	402, 462, 532, 612,	\section 46	\tikzset 96,
	955, 957, 958, 996,	\seq . 1375, 1376, 1377,	143, 209, 404, 430,
	997, 1107, 1108,	1504, 1505, 1506, 1513	1165, 1302, 1905,
	1111, 1113, 1115,	\setbox 2175	2031, 2235, 2389, 2596
	1120, 1129, 1136,	\setcounter	\tikzumlsset 2524
	1147, 1148, 1157,	1119, 2129, 2151, 2165	\times 207
	1161, 1162, 1163,	\setganttlinklabel ..	\tiny 966, 972,
	1191, 1192, 1231,	1149, 1150, 1151, 1152	978, 1370, 1424, 2334
	1300, 1384, 1389,	\setlength	\titleformat .. 1116, 1118
	1390, 1406, 1413,	2197, 2642, 2643, 2653	\titlespacing 1117
	1414, 1415, 1443,	\setmainfont 1109	\tl . 29, 37, 63, 64, 65,
	1558, 1719, 1720,	\setmainlanguage 396	66, 67, 68, 71, 72,
	1865, 1867, 1957,	\setminted 2324, 2325	73, 74, 75, 77, 79,
	2017, 2019, 2171,	\setsansfont 1110	104, 105, 106, 107,
	2172, 2173, 2186,	\setul 2188	108, 109, 110, 113,
	2214, 2245, 2296,	\sffamily 476,	114, 115, 116, 117,
	2302, 2307, 2318,	1116, 1118, 1220, 2408	118, 119, 162, 163,
		\shoveleft 1837	164, 165, 166, 167,
		\shoveright 1840	168, 171, 172, 173,
		\Sigma 64, 105,	174, 175, 176, 177,
		163, 1020, 1021, 1083	285, 289, 307, 311,
		\sigma 499, 501, 502	312, 313, 316, 321,
		\SLASH 1425	322, 323, 334, 335,
		\small 1534	336, 337, 348, 354,
		\sort 2262	357, 360, 369, 383,
		\sortList 2261, 2270	536, 539, 544, 545,
		\square 464	553, 554, 557, 558,
			1061, 1082, 1083,

