

lehramt-informatik

Hermine Bschlangaul *

August 24, 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	Namensschema der Entwurfsmuster-Makros:	31
2.10.2	Reihenfolge	31
2.10.3	Abstrakte Fabrik (Abstract Factory)	31
2.10.4	Adapter	32
2.10.5	Beobachter (Observer)	34
2.10.6	Dekorierer (Decorator)	36
2.10.7	Einfache Fabrik (Simple Factory)	37
2.10.8	Einzelstück (Singleton)	38
2.10.9	Erbauer (Builder)	38
2.10.10	Fabrikmethode (Factory Method)	40
2.10.11	Kompositum (Composite)	41
2.10.12	Modell-Präsentation-Steuerung (Model-View-Controller)	42
2.10.13	Zustand (State)	43
2.11	er.sty	45
2.12	formale-sprachen.sty	47
2.13	formatierung.sty	51
2.13.1	Schriftarten / Typographie	51
2.13.2	Farben	51
2.13.3	Überschriften	51

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

2.13.4	Listen	51
2.13.5	Kasten	51
2.13.6	Header	51
2.14	gantt.sty	52
2.15	grafik.sty	53
2.16	graph.sty	54
2.17	hanoi.sty	56
2.18	komplexitaetstheorie.sty	57
2.19	kontrollflussgraph.sty	59
2.20	kopf-fusszeilen.sty	61
2.21	literatur-dummy.sty	62
2.22	literatur.sty	63
2.23	makros.sty	64
2.24	master-theorem.sty	68
2.25	mathe.sty	72
2.26	minimierung.sty	73
2.27	normalformen.sty	76
2.28	petri.sty	78
2.29	potenzmengen-konstruktion.sty	80
2.30	pseudo.sty	82
2.31	pumping-lemma.sty	83
2.32	quicksort.sty	84
2.33	relationale-algebra.sty	87
2.34	rmodell.sty	88
2.35	sortieren.sty	89
2.36	spalten.sty	91
2.37	struktogramm.sty	92
2.38	syntax.sty	93
2.39	syntaxbaum.sty	95
2.40	synthese-algorithmus.sty	96
2.41	tabelle.sty	99
2.42	typographie.sty	100
2.43	uml.sty	101
2.44	vollstaendige-induktion.sty	103
2.45	wasserfall.sty	105
2.46	wpkalkuel.sty	106

3 Index

107

1 Klassen

1.1 Vorlage Theorie-Teil

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

\begin{document}

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

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

1.2 Vorlage Aufgabensammlung

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

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

1.3 Vorlage Aufgabe

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

\end{document}
```

2 Pakete

2.1 aufgaben-einbinden.sty

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

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

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

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

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

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

18
```


2.2 aufgaben-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=z_0, 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-qtrees.]

401 \RequirePackage{tikz}

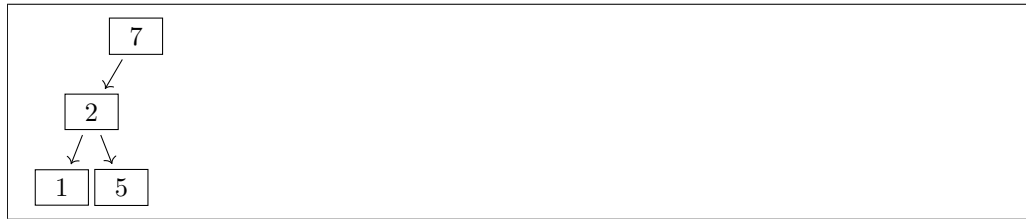
    für li binaer baum
402 \RequirePackage{tikz-qtrees}

    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}

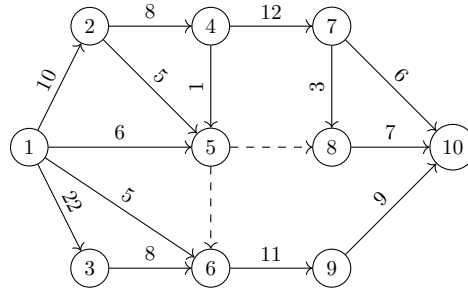
```

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

$R_a$ & $R_c$ & $R_b$ & $R_c$ & $R_a$ & $R_b$ \l6
B      & A      & A      & B      & C \l5
S      & -      & S      & S \l4
-      & -      & - \l3
-      & - \l2
S \l1
\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
610 Hilfsmakros zum Setzen von Entwurfsmustern / Design Patterns]
```

2.10.1 Namensschema der Entwurfsmuster-Makros:

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

2.10.2 Reihenfolge

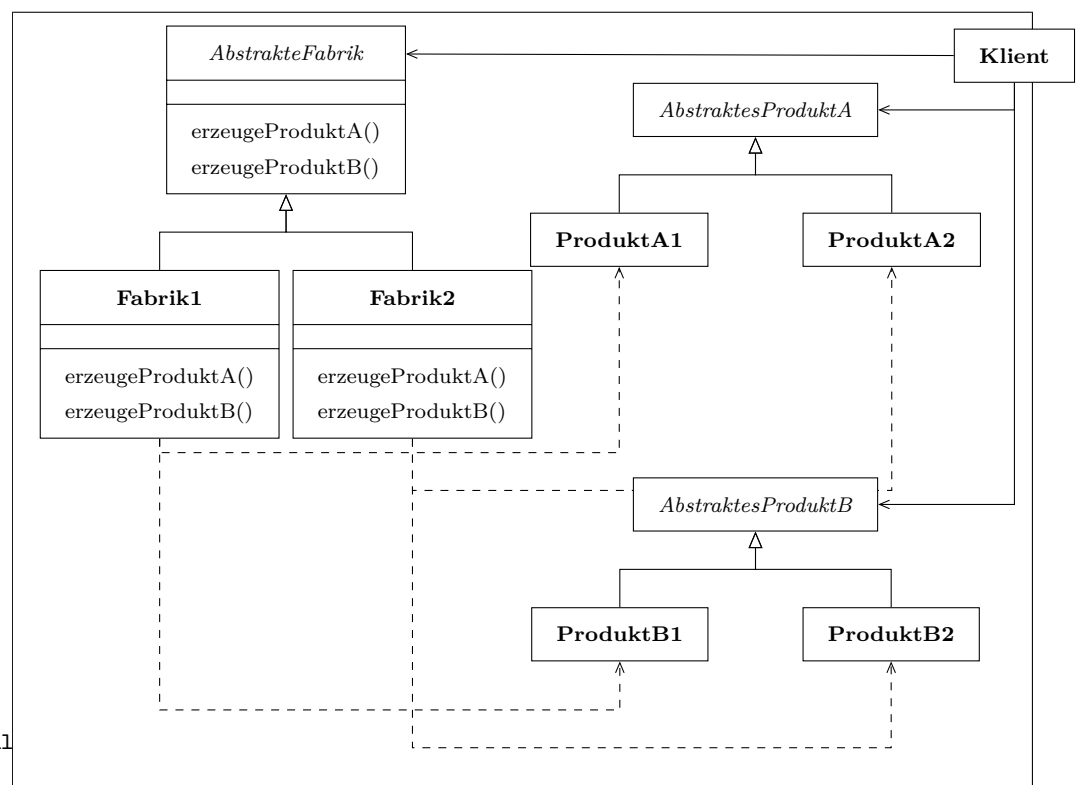
1. Uml: Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. Akteure: Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. Code: Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

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

```
612 \RequirePackage{lehramt-informatik-syntax}
613 \def\li@EntwurfsCode#1#2{
614   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
615 }
```

2.10.3 Abstrakte Fabrik (Abstract Factory)



```

616 \def\liEntwurfsAbstrakteFabrikUml{
617   \begin{tikzpicture}
618     \umlclass[type=abstract]{AbstrakteFabrik}{-}{
619       erzeugeProduktA()\backslash
620       erzeugeProduktB()\backslash
621     }

```

```

622 \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
623     erzeugeProduktA()\
624     erzeugeProduktB()\
625 }
626 \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
627     erzeugeProduktA()\
628     erzeugeProduktB()\
629 }
630 \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
631 \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
632
633 \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
634 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
635 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
636 \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
637 \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
638
639 \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
640
641 \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
642 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
643 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
644 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
645 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
646
647 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
648 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
649
650 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
651 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
652
653 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
654 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
655 \umlVHuniassoc{Klient}{AbstraktesProduktB}
656 \end{tikzpicture}
657 }

```

iEntwurfsAbstrakteFabrikCode

```

658 \def\liEntwurfsAbstrakteFabrikCode{
659     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
660     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
661     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
662 }

```

\liEntwurfsAbstrakteFabrik

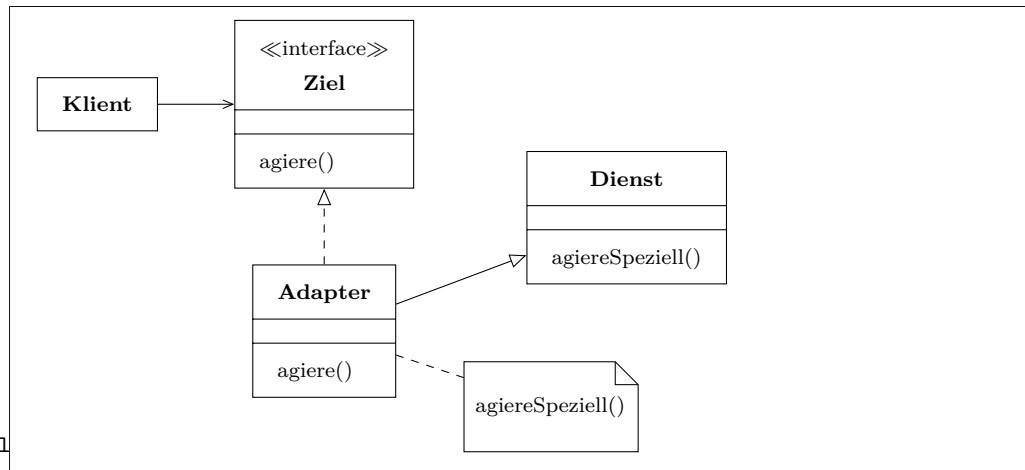
```

663 \def\liEntwurfsAbstrakteFabrik{
664     \liEntwurfsAbstrakteFabrikUml
665     \liEntwurfsAbstrakteFabrikCode
666 }

```

2.10.4 Adapter

\liEntwurfsAdapterUml



```

667 \def\liEntwurfsAdapterUml{
668   \begin{tikzpicture}
669     \umlsimpleclass[x=1,y=3]{Klient}{}{}
670     \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
671     \umlclass[x=4,y=0]{Adapter}{}{agiere()}
672     \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
673
674     \umlreal{Adapter}{Ziel}
675     \umluniassoc{Klient}{Ziel}
676     \umlinherit{Adapter}{Dienst}
677
678     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
679   \end{tikzpicture}
680   \footcite[so ähnlich wie GoF]{wiki:adapter}
681 }

```

\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.

```

682 \def\liEntwurfsAdapterAkteure{
683   \begin{description}
684     \item[Ziel (Target)]
685
686     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
687
688     \item[Klient (Client)]
689
690     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
691     dabei auf adaptierte Schnittstellen zurück.
692
693     \item[Dienst (Adaptee)]
694
695     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
696     definierter Schnittstelle an.
697
698     \item[Adapter]
699
700     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
701     Schnittstelle zum Klienten.\footcite{wiki:adapter}
702   \end{description}

```

```

703
704 \end{description}
705 }

```

\liEntwurfsAdapterCode

```

706 \def\liEntwurfsAdapterCode{
707 \li@EntwurfsCode{adapter}{Dienst}
708 \li@EntwurfsCode{adapter}{Ziel}
709 \li@EntwurfsCode{adapter}{Adapter}
710 \li@EntwurfsCode{adapter}{Klient}
711 }

```

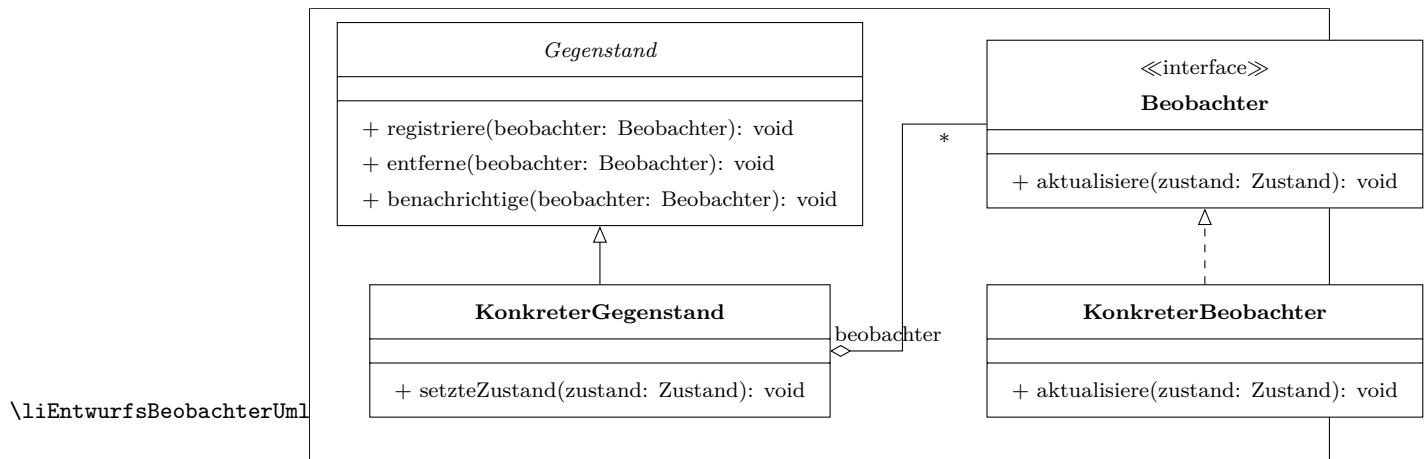
\liEntwurfsAdapter

```

712 \def\liEntwurfsAdapter{
713 \liEntwurfsAdapterUml
714 \liEntwurfsAdapterAkteure
715 \liEntwurfsAdapterCode
716 }

```

2.10.5 Beobachter (Observer)



```

717 \def\liEntwurfsBeobachterUml{
718 \begin{tikzpicture}
719 \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
720 + registriere(beobachter: Beobachter): void\\
721 + entferne(beobachter: Beobachter): void\\
722 + benachrichtige(beobachter: Beobachter): void\\
723 }
724 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
725 + setzteZustand(zustand: Zustand): void
726 }
727 \umlinherit{KonkreterGegenstand}{Gegenstand}
728
729 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
730 + aktualisiere(zustand: Zustand): void
731 }
732 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
733 + aktualisiere(zustand: Zustand): void
734 }
735 \umlreal{KonkreterBeobachter}{Beobachter}
736
737 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
738 {KonkreterGegenstand}{Beobachter}
739 \end{tikzpicture}
740 }

```

\liEntwurfsBeobachterAkteure

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

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

konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)

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

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

```
741 \def\liEntwurfsBeobachterAkteure{
742   \begin{description}
743     \item[Gegenstand / Subjekt (Subject / Observable)]
744
745     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
746     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
747     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
748     und Abmeldung von Beobachtern und eine Schnittstelle zur
749     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
750     251]{gof}
751
752     \item[Beobachter (Observer)]
753
754     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
755     genannt) definieren eine Aktualisierungsschnittstelle.
756
757     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
758
759     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
760     den relevanten Zustand und benachrichtigt alle Beobachter bei
761     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
762     verfügt über eine Schnittstelle zur Erfragung des aktuellen
763     Zustands.
764
765     \item[Konkrete Beobachter (ConcreteObserver)]
766
767     Konkrete Beobachter verwalten die Referenz auf ein konkretes
768     Subjekt, dessen Zustand sie beobachten und speichern und dessen
769     Zustand konsistent ist. Sie implementieren eine
770     Aktualisierungsschnittstelle unter Verwendung der
771     Abfrageschnittstelle des konkreten Subjekts.
772     \footcite{wiki:beobachter}
773   \end{description}
774 }
```

\liEntwurfsBeobachterCode

```
775 \def\liEntwurfsBeobachterCode{
776   \li@EntwurfsCode{beobachter}{Gegenstand}
777   \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
778   \li@EntwurfsCode{beobachter}{Beobachter}
779   \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
780   \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
```

```

781 \li@EntwurfsCode{beobachter}{Klient}
782 }

```

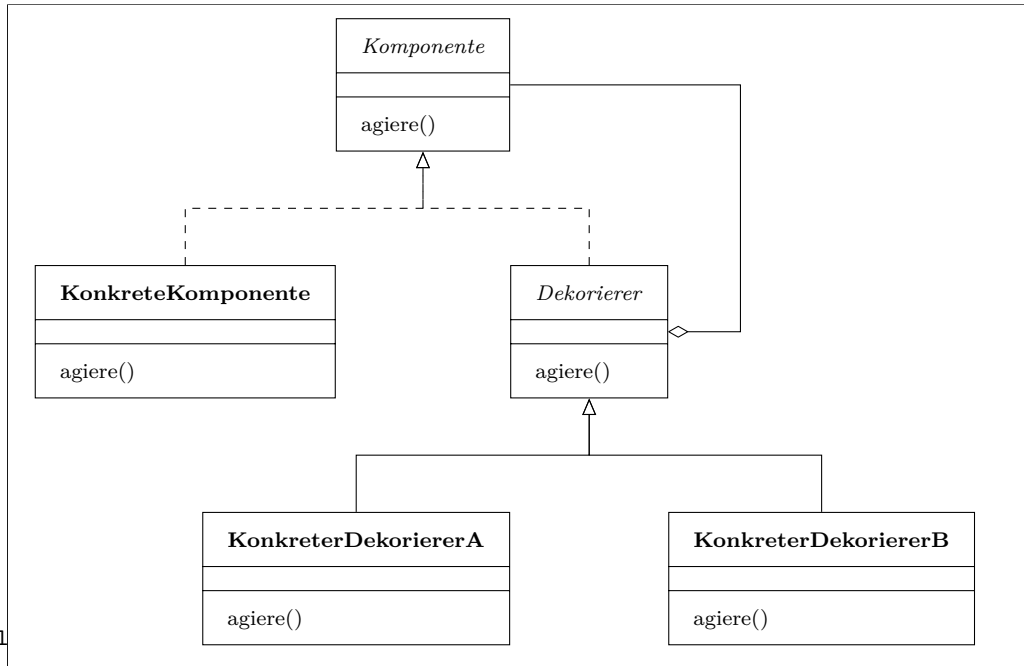
\liEntwurfsBeobachter

```

783 \def\liEntwurfsBeobachter{
784 \liEntwurfsBeobachterUml
785 \liEntwurfsBeobachterAkteure
786 \liEntwurfsBeobachterCode
787 }

```

2.10.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

788 \def\liEntwurfsDekoriererUml{
789 \begin{tikzpicture}
790 \umlclass[type=abstract]{Komponente}{-}{agiere()}
791 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
792 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
793
794 \umlVHVreal{KonkreteKomponente}{Komponente}
795 \umlVHVreal{Dekorierer}{Komponente}
796
797 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
798 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
799
800 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
801 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
802
803 \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
804 \footcite{wiki:dekorierer}
805 \end{tikzpicture}
806 }

```

\liEntwurfsDekoriererCode

```

807 \def\liEntwurfsDekoriererCode{
808 \li@EntwurfsCode{dekorierer}{Komponente}
809 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
810 \li@EntwurfsCode{dekorierer}{Dekorierer}
811 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
812 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
813 \li@EntwurfsCode{dekorierer}{Klient}
814 }

```

\liEntwurfsDekorierer

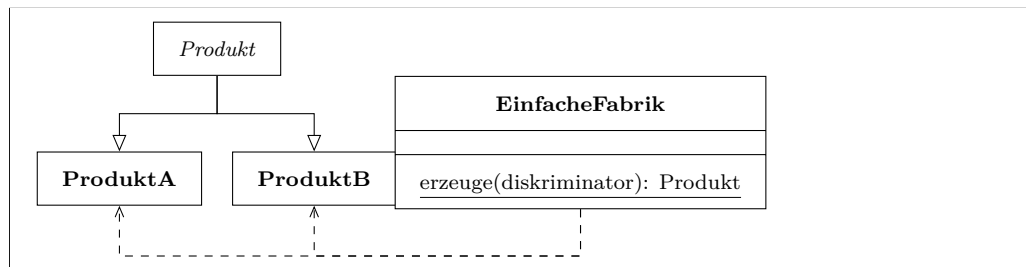
```

815 \def\liEntwurfsDekorierer{
816   \liEntwurfsDekoriererUml
817   \liEntwurfsDekoriererAkteure
818   \liEntwurfsDekoriererCode
819 }

```

2.10.7 Einfache Fabrik (Simple Factory)

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



```

820 \def\liEntwurfsEinfacheFabrikUml{
821   \begin{tikzpicture}
822     \umlsimpleclass[type=abstract]{Produkt}
823     \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
824     \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
825     \umlVHVinherit{Produkt}{ProduktA}
826     \umlVHVinherit{Produkt}{ProduktB}
827     \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
828     }{
829       \umlstatic{erzeuge(diskriminator): Produkt}\\
830     }
831     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
832     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
833   \end{tikzpicture}
834 }

```

\liEntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

835 \def\liEntwurfsEinfacheFabrikAkteure{
836   \begin{description}
837     \item[EinfacheFabrik]
838
839     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
840     Bedingung verschiedene Objekt instanziiert.
841
842     \item[Produkt]
843
844     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
845
846     \item[KonkretesProdukt]
847
848     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
849   \end{description}
850 }

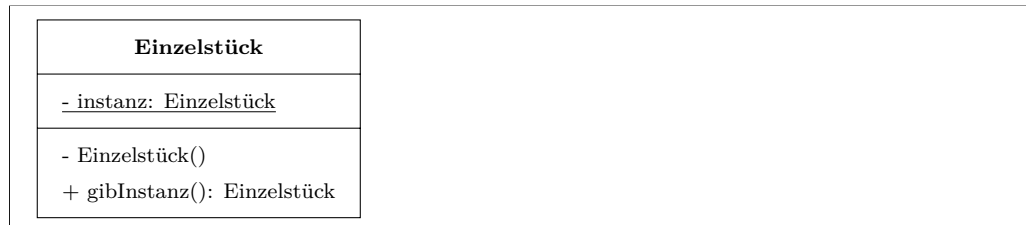
```

\liEntwurfsEinfacheFabrik

```
851 \def\liEntwurfsEinfacheFabrik{
852   \liEntwurfsEinfacheFabrikUml
853   \liEntwurfsEinfacheFabrikAkteure
854 }
```

2.10.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```
855 \def\liEntwurfsEinzelstueckUml{
856   \begin{tikzpicture}
857     \umlclass{Einzelstück}{
858       \umlstatic{- instanz: Einzelstück}\\
859     }{
860       - Einzelstück()\\
861       + gibInstanz(): Einzelstück
862     }
863   \end{tikzpicture}
864 }
```

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.

```
865 \def\liEntwurfsEinzelstueckAkteure{
866   \begin{description}
867     \item[Einzelstück (Singleton)]
868
869     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
870     nur auf eine einzige Instanz der Klasse zugreifen können.
871   \end{description}
872 }
```

\liEntwurfsEinzelstueckCode

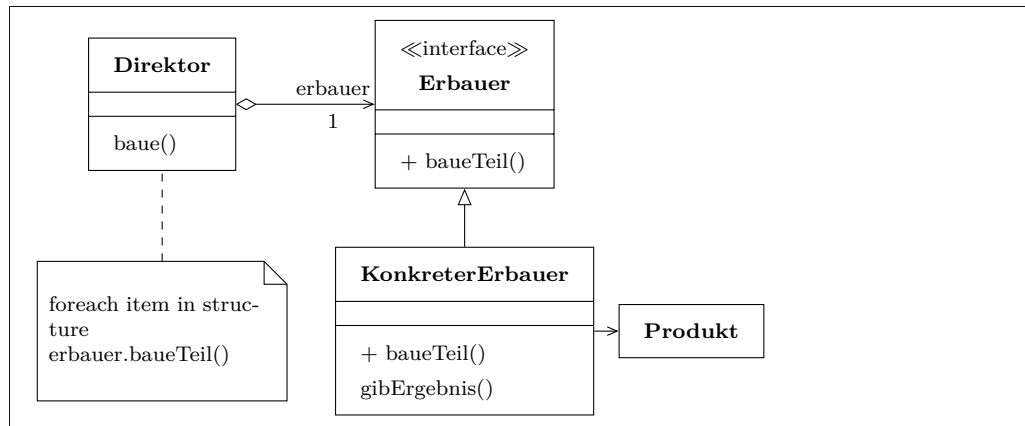
```
873 \def\liEntwurfsEinzelstueckCode{
874   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
875 }
```

\liEntwurfsEinzelstueck

```
876 \def\liEntwurfsEinzelstueck{
877   \liEntwurfsEinzelstueckUml
878   \liEntwurfsEinzelstueckAkteure
879   \liEntwurfsEinzelstueckCode
880 }
```

2.10.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

881 \def\liEntwurfsErbauerUml{
882   \begin{tikzpicture}
883     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
884     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
885     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
886       + baueTeil()\
887       gibErgebnis()}
888     \umlclass[x=0,y=3]{Direktor}{}{baue()}
889
890     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
891     \umluniassoc{KonkreterErbauer}{Produkt}
892     \umlinherit{KonkreterErbauer}{Erbauer}
893
894     \umlnote[x=0,y=0,width=3cm]{Direktor}{
895       foreach item in structure\
896       erbauer.baueTeil()
897     }
898   \end{tikzpicture}
899   \footcite{wiki:erbauer}
900 }

```

\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.

```

901 \def\liEntwurfsErbauerAkteure{
902   \begin{description}
903     \item[Erbauer]
904
905     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
906     Teile eines komplexen Objektes.
907
908     \item[KonkreterErbauer]
909
910     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
911     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er

```

```

912     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
913     Schnittstelle zum Auslesen des Produkts.
914
915     \item[Direktor]
916
917     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
918     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
919     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
920     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
921     Klienten.
922
923     \item[Produkt]
924
925     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
926     \footcite{wiki:erbauer}
927 \end{description}
928 }

```

\liEntwurfsErbauer

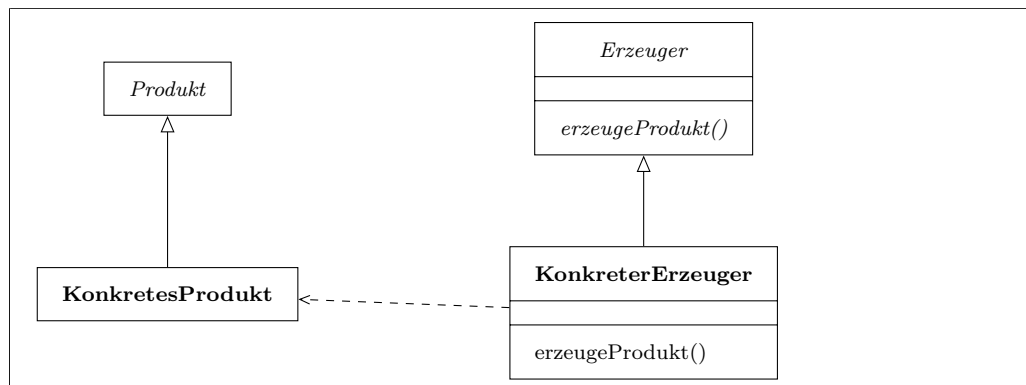
```

929 \def\liEntwurfsErbauer{
930   \liEntwurfsErbauerUml
931   \liEntwurfsErbauerAkteure
932 }

```

2.10.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

933 \def\liEntwurfsFabrikmethodeUml{
934   \begin{tikzpicture}
935     \umlsimpleclass[type=abstract]{Produkt}
936     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
937     \umlinherit{KonkretesProdukt}{Produkt}
938
939     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
940       \textit{erzeugeProdukt()}\}
941     }
942     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
943       erzeugeProdukt()
944     }
945     \umlinherit{KonkreterErzeuger}{Erzeuger}
946
947     \umldep{KonkreterErzeuger}{KonkretesProdukt}
948   \end{tikzpicture}
949 }

```

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).

```

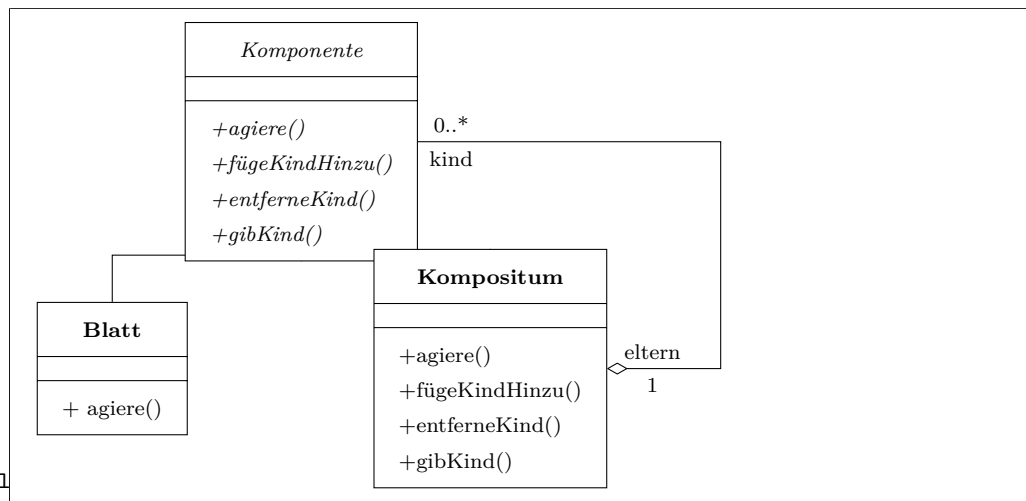
950 \def\liEntwurfsFabrikmethodeAkteure{
951   \begin{description}
952     \item[Produkt]
953
954     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
955     zu erzeugende Produkt.
956
957     \item[KonkretesProdukt]
958
959     KonkretesProdukt implementiert die Produkt-Schnittstelle.
960
961     \item[Erzeuger]
962
963     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
964     zu erzeugen und kann eine Default-Implementierung beinhalten.
965
966     \item[KonkreterErzeuger]
967
968     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
969     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
970     Konstruktor einer konkreten Produkt-Klasse aufruft).
971
972     \footcite{wiki:fabrikmethode}
973   \end{description}
974 }
```

\liEntwurfsFabrikmethode

```

975 \def\liEntwurfsFabrikmethode{
976   \liEntwurfsFabrikmethodeUml
977   \liEntwurfsFabrikmethodeAkteure
978 }
```

2.10.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

979 \def\liEntwurfsKompositumUml{
980   \begin{tikzpicture}
981     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
982       \textit{+agiere()}\
983       \textit{+fügeKindHinzu()}\
984       \textit{+entferneKind()}\
985       \textit{+gibKind()}
986     }
987     \umlclass[x=0]{Blatt}{-}{+ agiere()}
988     \umlclass[x=5]{Kompositum}{-}{-}{
989       +agiere()\
990       +fügeKindHinzu()\
991       +entferneKind()\
992       +gibKind()
993     }
994
995     \umlVHVinherit{Kompositum}{Komponente}
996     \umlVHVinherit{Blatt}{Komponente}
997     \umlHVVhaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,
998   \end{tikzpicture}
999 }

```

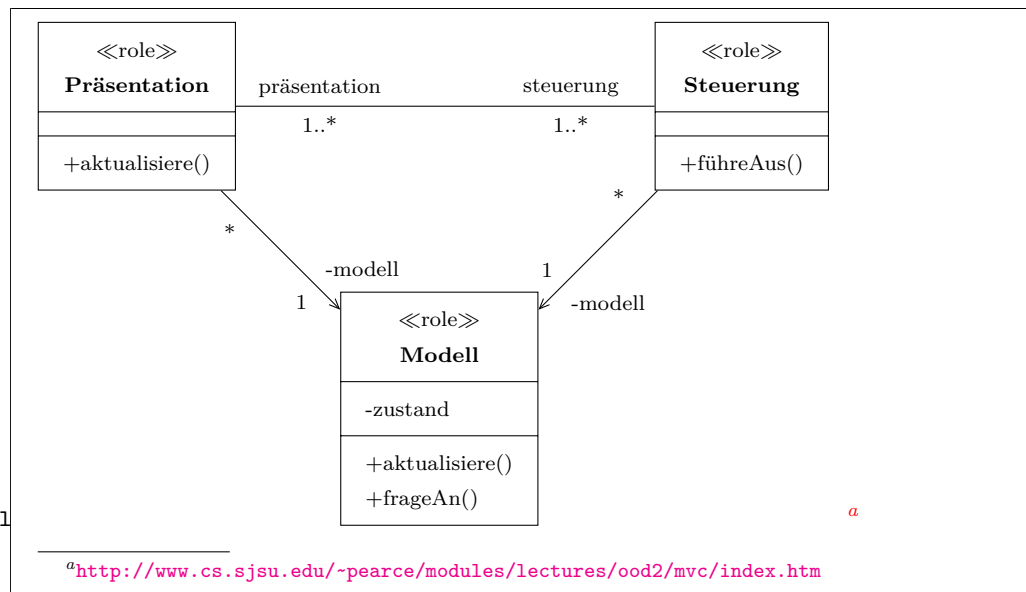
\liEntwurfsFabrikmethode

```

1000 \def\liEntwurfsKompositum{
1001   \liEntwurfsKompositumUml
1002   \liEntwurfsKompositumAkteure
1003 }

```

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



```

1004 \def\liEntwurfsModellPraesentationSteuerungUml{
1005   \begin{tikzpicture}
1006     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1007     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1008     \umlclass[y=-4cm,type=role]{Modell}{-}{
1009       -zustand
1010     }{
1011       +aktualisiere()\
1012       +frageAn()
1013     }
1014
1015     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1016     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1017 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1018 \end{tikzpicture}
1019 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1020 }

```

ModellPraesentationSteuerung

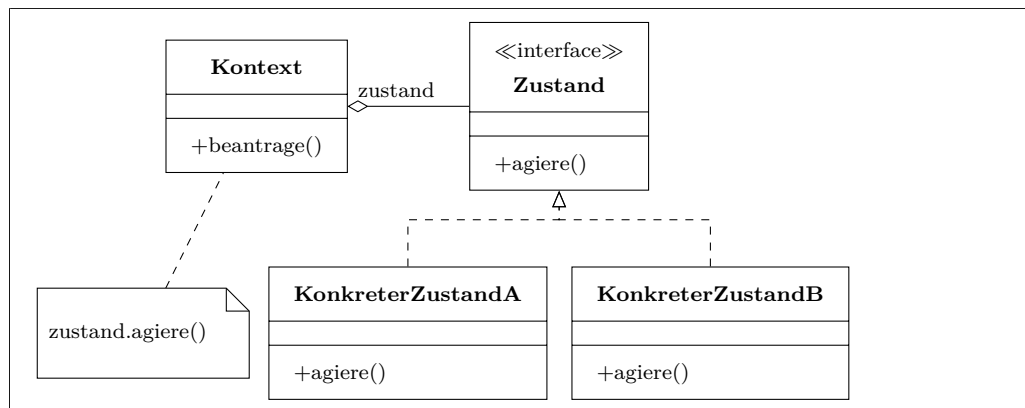
```

1021 \def\liEntwurfs{
1022 \liEntwurfsModellPraesentationSteuerungUml
1023 \liEntwurfsModellPraesentationSteuerungAkteure
1024 }

```

2.10.13 Zustand (State)

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



```

1025 \def\liEntwurfsZustandUml{
1026 \begin{tikzpicture}
1027 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1028 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1029 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1030 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1031
1032 \umlVHVreal{KonkreterZustandA}{Zustand}
1033 \umlVHVreal{KonkreterZustandB}{Zustand}
1034
1035 \umlagg[reg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1036
1037 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1038 \end{tikzpicture}
1039 }

```

\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.

```

1040 \def\liEntwurfsZustandAkteure{
1041 \begin{description}
1042 \item[Kontext (Context)]
1043
1044 definiert die clientseitige Schnittstelle und verwaltet die separaten
1045 Zustandsklassen.
1046

```

```

1047 \item[State (Zustand)]
1048
1049 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1050 implementiert gegebenenfalls ein Standardverhalten.
1051
1052 \item[KontreterZustand (ConcreteState)]
1053
1054 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1055 verbunden ist.
1056 \end{description}
1057 }

\liEntwurfsZustand
1058 \def\liEntwurfsZustand{
1059 \liEntwurfsZustandUml
1060 \liEntwurfsZustandAkteure
1061 }

1062

```

2.11 er.sty

```
1063 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1064 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1065 ER-Diagrammen]

1066 \RequirePackage{tikz-er2}
1067 \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}

1068 \RequirePackage{soul}
1069 \RequirePackage{fontawesome}

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

1070 \ExplSyntaxOn

\liErEntity
1071 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
1072 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
1073 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\ e=\liErMpEntity
1074 \def\liErMpEntity#1{
1075   \liErEntity{#1}
1076   \marginpar{
1077     \liErEntity{\tiny\faSquareO{}}~E:~#1}
1078   }
1079 }
□

\liErMpRelationship Let-Abkürzung: \let\ r=\liErMpRelationship
1080 \def\liErMpRelationship#1{
1081   \liErRelationship{#1}
1082   \marginpar{
1083     \liErRelationship{\tiny\faGg{}}~R:~#1}
1084   }
1085 }

\liErMpAttribute Let-Abkürzung: \let\ a=\liErMpAttribute
1086 \def\liErMpAttribute#1{
1087   \liErAttribute{#1}
1088   \marginpar{
1089     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1090   }
1091 }

\liErDatenbankName Let-Abkürzung: \let\ d=\liErDatenbankName
datenbank name
1092 \def\liErDatenbankName#1{
1093   {
1094     \footnotesize\texttt{(#1)}
1095   }
1096 }

1097 \ExplSyntaxOff
1098

```

2.12 formale-sprachen.sty

```

1099 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1100 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1101 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
1102
1103 \directlua{
1104   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1105 }
1106
1107 \RequirePackage{hyperref}
1108 \RequirePackage{lehramt-informatik-mathe}

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

1109 \def\liMengeOhneMathe#1{\{ #1 \}}
1110 \def\liMenge#1{%
1111   \ifmmode%
1112     \liMengeOhneMathe{#1}%
1113   \else%
1114     $\liMengeOhneMathe{#1}$%
1115   \fi%
1116 }

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

1117 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1118 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1119 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1120 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1121 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1122 \let\liZustandsmenge=\liPotenzmenge

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

1123 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1124 \def\liUeberfuehrungsFunktion#1{
1125   \ifmmode
1126     \liUeberfuehrungsFunktionOhneMathe{#1}
1127   \else
1128     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1129   \fi
1130 }

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

1131 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1132 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1133 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1134 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1135 \def\zustandsnamens@liste#1#2{
1136   $
1137   \{
1138     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1139   \}
1140   $
1141 }
1142 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1143 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1144 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1145 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1146 \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}

1147 \NewDocumentEnvironment { liProduktionsRegeln }
1148 { 0{P} +b }
1149 {
1150   \noindent
1151   $#1 = \{ $
1152   \vspace{-0.2cm}
1153   \begin{align*}
1154     \directlua{formale_sprachen.produktions_regeln('#2')}
1155   \end{align*}
1156   \vspace{-1.5cm}
1157   \begin{flushright}\}\end{flushright}
1158 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1159 \def\liProduktionen#1{
1160   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1161 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1162 \def\liZustandsnameTiefgestellt#1{
1163   \ifmmode
1164     \liZustandsBuchstabe\sb{#1}
1165   \else
1166     $\liZustandsBuchstabe\sb{#1}$
1167   \fi
1168 }

1169 \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}

1170 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1171   $
1172   \tl_if_empty:nTF {#1} {} {#1 =}
1173   \{
1174     \, #2 \,
1175     |
1176     \, #3 \,
1177   \}$
1178 }
1179 \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

1180 \def\liFlaci#1{%
1181   \par
1182   {%
1183     \scriptsize
1184     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1185     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1186     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1187     \href{https://flaci.com/#1}{flaci.com/#1}%
1188   }%
1189   \par
1190 }

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

1191 \ExplSyntaxOn
1192 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1193   \tl_set:Nn \l_variablen_tl {V}
1194   \tl_set:Nn \l_alphabet_tl {\Sigma}
1195   \tl_set:Nn \l_produktionen_tl {P}
1196   \tl_set:Nn \l_start_tl {S}
1197
1198   \keys_define:nn { grammatik } {
1199     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1200     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1201     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1202     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1203   }
1204
1205   \keys_set:nn { grammatik } { #2 }
1206
1207   $#1 = (
1208     \l_variablen_tl,
1209     \l_alphabet_tl,
1210     \l_produktionen_tl,
1211     \l_start_tl
1212   )$

```

```
1213 }  
1214 \ExplSyntaxOff  
  
1215
```

2.13 formatierung.sty

```
1216 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1217 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.13.1 Schriftarten / Typographie

```
1218 \RequirePackage{mathpazo}
1219 \RequirePackage[no-math]{fontspec}
1220 \setmainfont{texgyrepagella}
1221 \setsansfont{QTAncientOlive}
1222 \RequirePackage{sectsty}
1223 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.13.2 Farben

```
1224 \RequirePackage{xcolor}
1225 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.13.3 Überschriften

```
1226 \RequirePackage{titlesec}
1227 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1228 \titlespacing{\chapter}{0pt}{0pt}{*1}
1229 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1230 \setcounter{secnumdepth}{0}
```

2.13.4 Listen

```
1231 \RequirePackage{paralist}
1232 \renewcommand\labelitemi{-}
1233 \renewcommand\labelitemii{-}
1234 \renewcommand\labelitemiii{-}
1235 \renewcommand\labelitemiv{-}
1236 % Counter: enumi enumii enumiii enumiv
1237 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1238 \renewcommand{\labelenumi}{(\alph{enumi})}
1239 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.13.5 Kasten

```
1240 \RequirePackage{mdframed}
1241 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1242 \NewDocumentEnvironment { liKasten }{ } {
1243   \begin{mdframed}[backgroundcolor=white!0]
1244 } {
1245   \end{mdframed}
1246 }
```

2.13.6 Header

```
1247 \RequirePackage{fancyhdr}
1248 \fancyhead[L,C,R]{}
1249 \fancyfoot[L]{}
1250 \fancyfoot[C]{}
1251 \fancyfoot[R]{\thepage}
1252 \pagestyle{fancy}
1253 \renewcommand{\headrulewidth}{0pt}
1254 \renewcommand{\footrulewidth}{0pt}
1255
```

2.14 gantt.sty

```

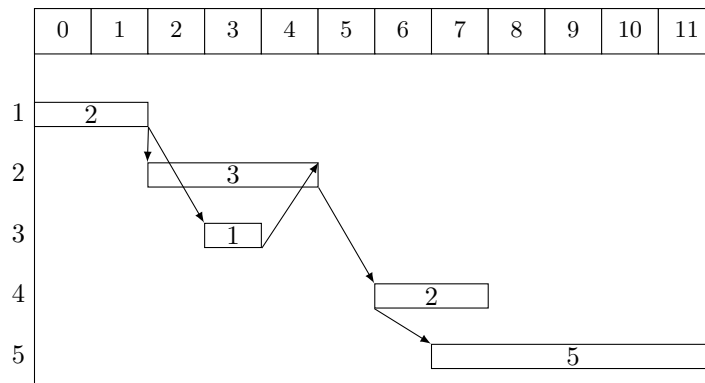
1256 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1257 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1258 \RequirePackage{tikz-uml}
1259 \RequirePackage{pgfgantt}
1260 \setganttlinklabel{f-s}{}
1261 \setganttlinklabel{s-s}{}
1262 \setganttlinklabel{f-f}{}
1263 \setganttlinklabel{s-f}{}
1264

```

2.15 grafik.sty

```
1265 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1266 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1267 TikZ, um Grafiken zeichnen zu können.]
1268 \RequirePackage{tikz}
1269
```

2.16 graph.sty

```

1270 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1271 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1272 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1273 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1274 \RequirePackage{blkarray}
1275 \usetikzlibrary{arrows.meta}

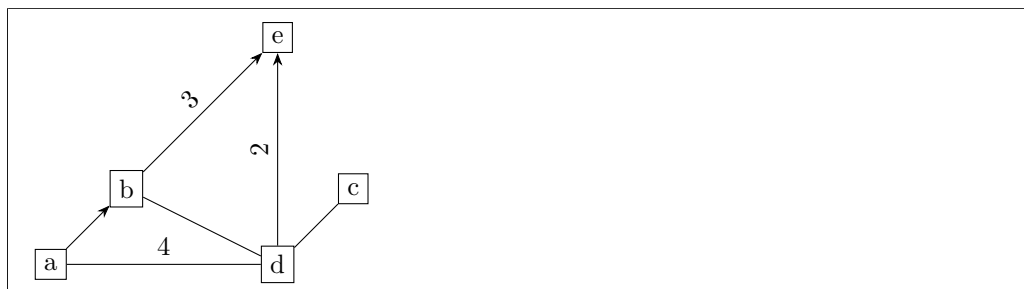
```

```

\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}

```



```

1276 \tikzset{
1277   li graph/.style={
1278     every node/.style={
1279       rectangle,
1280       draw,
1281     },
1282     every edge/.style={
1283       >={Stealth[black]},
1284       draw,
1285     },
1286     every edge/.append style={
1287       every node/.style={
1288         sloped,
1289         auto,
1290       }
1291     }
1292   },
1293   li markierung/.style={
1294     ultra thick,
1295   }
1296 }

```

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}

```

```

1297 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1298

```

2.17 hanoi.sty

1299 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1300 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
 1301 von Hanoi-Grafiken]

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

```

1302 \RequirePackage{tikz}
1303 \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}

1304 \def\li@mset #1[#2]=#3{%
1305 \expandafter\xdef\csname #1#2\endcsname{#3}
1306 }
1307 \def\li@mget #1[#2]{%
1308 \csname #1#2\endcsname
1309 }
1310 \def\li@minc #1[#2]+=#3{%
1311 \pgfmathparse{\li@mget #1[#2]+#3}%
1312 \li@mset #1[#2]=\pgfmathresult
1313 }
1314
1315 \def\liHanoi#1#2{
1316   \edef\li@numdiscs{#1}
1317   \def\li@sequence{#2}
1318   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1319     % init colors
1320     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1321       \li@mset col[\j]={\c};
1322     }
1323     % draw poles and init pole counters
1324     \foreach \j in {1,2,3}{
1325       \li@mset pos[\j]=0
1326       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1327     }
1328     % draw base
1329     \draw (.5,-.5) -- +(3,0);
1330     % draw discs
1331     \foreach[count=\k] \i/\j in \li@sequence{
1332       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1333       \li@minc pos[\j]+=.5
1334     }
1335   \end{tikzpicture}
1336 }
```


2.18 komplexitaetstheorie.sty

```

1337 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1338 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1339 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1340 Polynomialzeitreduktion.]

```

Let-Abkürzungen

```

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

```

```

1341 \liLadePakete{mathe}

```

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

```

1342 \RequirePackage{mdframed}

```

```

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

```

```

1343 \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

```

```

1344 \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`

```

1345 \def\liProblemBeschreibung#1#2#3{
1346   \begin{mdframed}[
1347     userdefinedwidth=9cm,
1348     align=center,
1349     backgroundcolor=white!0,
1350   ]
1351   \centerline{\large\liProblemName{#1}}
1352
1353   \medskip
1354
1355   \begin{description}
1356     \item[Gegeben:] #2
1357     \item[Frage:] #3
1358   \end{description}
1359   \end{mdframed}
1360 }

```

```

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

1361 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1362 \begin{displaymath}
1363 \liProblemName{#1}
1364 \preceq_{#2}
1365 \liProblemName{#3}
1366 \end{displaymath}
1367 }

\liProblemVertexCover

1368 \def\liProblemClique{%
1369 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1370 Mindestgröße  $n$  in einem gegebenen Graphen.
1371 \footcite[Seite 76]{theo:fs:4}
1372 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1373 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1374 \footcite{wiki:cliquenproblem}
1375 }

\liProblemVertexCover

1376 \def\liProblemVertexCover{%
1377 %
1378 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1379 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1380 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1381 \footcite{wiki:knotenueberdeckung}
1382
1383 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1384 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1385 mindestens einem Knoten aus  $U$  verbunden ist.
1386 \footcite[Seite 78]{theo:fs:4}%
1387 }

\liProblemSubsetSum Kein Karp-21-Problem

1388 \def\liProblemSubsetSum{%
1389 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1390 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1391 \footcite{wiki:teilsommenproblem}
1392 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1393 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1394 größer als eine gegebene obere Schranke  $c$  ist.
1395 \footcite[Seite 74]{theo:fs:4}
1396 }

\liProblemSubsetSum Kein Karp-21-Problem

1397 \def\liProblemSat{%
1398 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1399 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1400 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1401 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1402 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1403 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1404 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1405 aufgestellt werden.
1406 \footcite[Seite 71]{theo:fs:4}
1407 }

1408

```

2.19 kontrollflussgraph.sty

```
1409 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1410 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1411 \RequirePackage{tikz}
1412 \usetikzlibrary{positioning}
1413 \tikzset{
1414   li kontrollfluss/.style={
1415     knoten/.style={
1416       circle,
1417       draw
1418     },
1419     usebox/.style={
1420       draw,
1421       rectangle,
1422       font=\scriptsize,
1423       anchor=west,
1424       align=left,
1425     },
1426     bedingung/.style={
1427       midway,
1428       draw=none,
1429       font=\scriptsize
1430     },
1431     knotenbeschriftung/.style={
1432       draw,
1433       rectangle,
1434       midway,
1435       font=\scriptsize
1436     },
1437     wahr/.style={
1438       thick
1439     },
1440     falsch/.style={
1441       dashed
1442     },
1443     every node/.style={
1444       circle,
1445       draw,
1446     },
1447     every edge/.append style={
1448       every node/.style={
1449         draw=none,
1450         bedingung,
1451       }
1452     },
1453     every path/.style={
1454       draw,
1455       ->,
1456     },
1457     every pin/.style={
1458       draw,
1459       dotted,
1460       rectangle,
1461       pin position=right
1462     },
1463     every pin edge/.style={
1464       dotted,
1465       arrows=-,
1466     }
1467   }
1468 }
```

liKontrollflussgraph

```

1469 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {
1470   \begin{tikzpicture}[
1471     li kontrollfluss,
1472     #1
1473   ]
1474 } {
1475   \end{tikzpicture}
1476 }

\liAnweisung
1477 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1478 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1479 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1480 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1481 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1482 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1483 \ExplSyntaxOn
1484 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1485 {
1486   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1487   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1488   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1489 }
1490 \ExplSyntaxOff

1491

```

2.20 kopf-fusszeilen.sty

```
1492 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1493 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1494 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1495 \ExplSyntaxOn

1496 \fancyhead{}
1497 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1498 \fancyhead[L0,RE]{\scriptsize\today}
1499 \fancyfoot{}
1500 \fancyfoot[LE,R0]{\thepage}
1501 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1502 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1503 \renewcommand{\headrulewidth}{0.4pt}
1504 \renewcommand{\footrulewidth}{0.4pt}

1505 \ExplSyntaxOff

1506
```

2.21 literatur-dummy.sty

```
1507 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1508 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1509 \def\literatur{}

\footcite

1510 % \RequirePackage[stable,multiple]{footmisc}
1511 \NewDocumentCommand{ \footcite } { o m }{}

1512
```

2.22 literatur.sty

```
1513 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1514 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1515 \RequirePackage{csquotes}
1516 \RequirePackage[
1517   bibencoding=utf8,
1518   citestyle=authortitle,
1519   backend=biber,
1520 ]{biblatex}
1521 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1522 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1523 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1524 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1525 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1526 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1527 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1528 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1529 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1530 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1531 % To allow footnotes in the heading
1532 \RequirePackage[stable,multiple]{footmisc}

\literatur

1533 \def\literatur{\printbibliography[heading=subbibliography]}

1534
```

2.23 makros.sty

```

1535 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1536 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1537 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1538 anderen Paket passen]
1539 \RequirePackage{hyperref}
1540 \RequirePackage{graphicx}

Für die Umgebung liQuellen benötigt.
1541 \RequirePackage{paralist}

\inhaltsverzeichnis
1542 \def\inhaltsverzeichnis {
1543   \begin{mdframed}
1544     \begin{group}
1545       \let\clearpage\relax
1546       \tableofcontents
1547     \end{group}
1548   \end{mdframed}
1549 }

\memph \mephm (\marginpar and \emph)
1550 \newcommand\memph{1}{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1551 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1552 \newcommand\liPseudoUeberschrift{1}{
1553   \bigskip
1554   \noindent
1555   \textsf{\textbf{#1}}
1556   \noindent
1557 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1558 \newcommand\liBeschriftung{1}{
1559   \par
1560   \noindent
1561   \medskip
1562   \textbf{#1}:
1563   \medskip
1564   \noindent
1565 }

\hinweis
1566 \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.
1567 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1568 \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.
1569 \RequirePackage{xparse}
1570 \ExplSyntaxOn

```



```

1571 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1572 {
1573   \str_case:nn {#1} {
1574     {standard} {
1575       \def\beschriftung{}
1576       \mdfsetup{backgroundcolor=gray!5,linecolor=gray}
1577     }
1578     {richtig} {
1579       \def\beschriftung{richtig}
1580       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1581     }
1582     {falsch} {
1583       \def\beschriftung{falsch}
1584       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1585     }
1586     {muster} {
1587       \def\beschriftung{Musterlösung}
1588       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1589     }
1590   }
1591   \ifx\beschriftung\empty\else
1592     \noindent
1593     \textbf{\beschriftung{:}}
1594   \fi
1595   \begin{mdframed}
1596 }
1597 {\end{mdframed}}

```

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

```

1598 \NewDocumentEnvironment{ liAdditum } { o }
1599 {
1600   \begin{mdframed}[backgroundcolor=yellow!5]
1601     \IfNoValueTF {#1}
1602     { \liPseudoUeberschrift{Additum} }
1603     { \liPseudoUeberschrift{Additum:~#1} }
1604   }
1605 {\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.

```

1606 \NewDocumentEnvironment{ liExkurs }{o +b}{
1607   \vspace{0.2cm}%
1608   \begin{mdframed}[
1609     backgroundcolor=white,
1610     bottomline=false,
1611     innermargin=1cm,
1612     leftline=true,
1613     linecolor=black,
1614     linewidth=0.1cm,
1615     outermargin=1cm,
1616     rightline=false,
1617     topline=false,
1618   ]

```

```

1619 \footnotesize
1620 \noindent%
1621 \textbf{Exkurs:~#1}\par%
1622 \noindent%
1623 #2
1624 \end{mdframed}
1625 \vspace{0.2cm}
1626 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1627 \cs_new:Npn \listen@punkt #1 {\item #1}
1628 \NewDocumentEnvironment { liQuellen }{ +b }
1629 {
1630   \seq_clear_new:N \l_quellen
1631   \seq_set_split:Nnn \l_quellen {\item} {#1}
1632   \seq_remove_all:Nn \l_quellen {}
1633   \begin{mdframed}[backgroundcolor=white!0]
1634     \footnotesize
1635     \noindent
1636     \textsf{\textbf{Weiterführende~Literatur:}}
1637     \medskip
1638     \begin{compactitem}
1639       \seq_map_function:NN \l_quellen {\listen@punkt}
1640     \end{compactitem}
1641   \end{mdframed}
1642   %
1643   \makeatletter
1644   \par\@afterindentfalse\@afterheading
1645   \makeatother
1646 } {}

```

liLernkartei

```

1647 \NewDocumentEnvironment { liLernkartei }{ m +b }
1648 {
1649   \begin{mdframed}
1650     \footnotesize
1651     \noindent%
1652     \textbf{Lernkarteikarte:~#1}\par%
1653     \noindent%
1654     #2
1655   \end{mdframed}
1656 } {}

```

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

```

1657 \NewDocumentEnvironment { liDiagramm }{ m +b }
1658 {
1659   \begin{mdframed}[backgroundcolor=white!0]
1660     \small
1661     \noindent%
1662     \textit{#1}:
1663     \begin{center}

```

```

1664 #2
1665 \medskip
1666 \end{center}
1667 \end{mdframed}
1668 } {}
1669 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1670 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1671 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1672 }
1673

\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.
1674 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1675 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1676 }

\zB
1677 \def\zB{z.\,B. }

\ZB
1678 \def\ZB{Z.\,B. }

\dh
1679 \def\dh{d.\,h. }

1680

```

2.24 master-theorem.sty

1681 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1682 \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{0}(1)} % f(n)

\liMasterFallRechnung

% 1. Fall

{für $\varepsilon = 4$: \}

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

% 2. Fall

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

% 3. Fall

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

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

\liMasterVariablenDeklaration

{ } % a

{ } % b

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

\liMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

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

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

1683 \ExplSyntaxOn

1684 \RequirePackage{amsmath}

\liRundeKlammer

1685 \def\liRundeKlammer#1{

1686 \negthinspace \left(#1 \right)

1687 }

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

1688 \def\liThetaOhneMathe#1{

1689 \Theta \liRundeKlammer{#1}

1690 }

1691 \def\liTheta#1{

1692 \ifmmode

1693 \liThetaOhneMathe{#1}

1694 \else

1695 \$\liThetaOhneMathe{#1}\$

1696 \fi

1697 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1698 \def\liOmegaOhneMathe#1{
1699 \Omega \liRundeKlammer{#1}
1700 }
1701 \def\liOmega#1{
1702 \ifmmode
1703 \liOmegaOhneMathe{#1}
1704 \else
1705 $\liOmegaOhneMathe{#1}$
1706 \fi
1707 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1708 \def\liOOhneMathe#1{
1709 \mathcal{O} \liRundeKlammer{#1}
1710 }
1711 \def\liO#1{
1712 \ifmmode
1713 \liOOhneMathe{#1}
1714 \else
1715 $\liOOhneMathe{#1}$
1716 \fi
1717 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1718 \def\liTOhneMathe#1#2{
1719 \tl_if_blank:nTF {#1}
1720 {}
1721 {#1 \cdot }
1722 T
1723 \liRundeKlammer{\frac{n}{#2}}
1724 }
1725 \def\liT#1#2{
1726 \ifmmode
1727 \liTOhneMathe{#1}{#2}
1728 \else
1729 $\liTOhneMathe{#1}{#2}$
1730 \fi
1731 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1732 \def\liRekursionsGleichung{
1733 $T(n) = \liT{a}{b} + f(n)$
1734 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1735 \def\liBedingungEins{
1736 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1737 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1738 \def\liBedingungZwei{
1739 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1740 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1741 \def\liBedingungDrei{
1742 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1743 }
1744 \ExplSyntaxOff

```

\liMasterVariablen

```

1745 \def\liMasterVariablen{
1746   \begin{displaymath}
1747     T(n) = \liT{a}{b} + f(n)
1748   \end{displaymath}
1749
1750   \begin{itemize}
1751     \item[$a = $]
1752       Anzahl der Unterprobleme in der Rekursion
1753
1754     \item[$\textstyle{\frac{1}{b}} = $]
1755       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1756       repräsentiert wird
1757
1758     \item[$f(n) = $]
1759       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1760       die Kombination der Teillösungen entstehen
1761   \end{itemize}
1762   \footcite{wiki:master-theorem}
1763   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1764 }
```

\liMasterFaelle

```

1765 \def\liMasterFaelle{
1766   \begin{description}
1767     \item[1. Fall:]
1768        $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1769
1770       \hfill falls \liBedingungEins
1771       für  $\varepsilon > 0$ 
1772
1773     \item[2. Fall:]
1774        $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1775
1776       \hfill falls \liBedingungZwei
1777
1778     \item[3. Fall:]
1779        $T(n) \in \liTheta{f(n)}$ 
1780
1781       \hfill falls \liBedingungDrei
1782       für  $\varepsilon > 0$ 
1783       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1784       gilt:
1785        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1786     \end{description}
1787 }
```

\liMasterVariablenDeklaration

```

1788 \def\liMasterVariablenDeklaration#1#2#3{
1789   \begin{description}
1790     \item[Allgemeine Rekursionsgleichung:] \strut
1791
1792     \liRekursionsGleichung
1793
1794     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1795
1796     #1
1797
1798     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1799
1800     um  $\frac{1}{\#2}$  also  $b = \#2$ 
1801
1802     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
```

```

1803
1804     $#3$
1805
1806     \item[Ergibt folgende Rekursionsgleichung:] \strut
1807
1808      $T(n) = T_{\#1\#2} + \#3$ 
1809     \end{description}
1810 }

\liMasterFallRechnung
1811 \def\liMasterFallRechnung#1#2#3{
1812     \begin{description}
1813     \item[1. Fall:] \liBedingungEins:
1814
1815         #1
1816
1817     \item[2. Fall:] \liBedingungZwei:
1818
1819         #2
1820
1821     \item[3. Fall:] \liBedingungDrei:
1822
1823         #3
1824     \end{description}
1825 }

\liMasterExkurs
1826 \def\liMasterExkurs{
1827     \begin{liExkurs}[Master-Theorem]
1828     \liMasterVariablen
1829
1830     \noindent
1831     Dann gilt:
1832
1833     \liMasterFaelle
1834     \end{liExkurs}
1835 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1836 \def\liMasterWolframLink#1{
1837     Berechne die Rekursionsgleichung auf WolframAlpha:
1838     \href{https://www.wolframalpha.com/input/?i=\#1\{WolframAlpha\}}
1839 }

1840

```

2.25 mathe.sty

```
1841 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1842 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1843
1844 % for example \ltimes \rtimes
1845 %\RequirePackage{amssymb}
1846 \RequirePackage{amsmath}
1847
1848 %%
1849 % \mlq \mrq
1850 %%
1851 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1852 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1853
```


2.26 minimierung.sty

```

1854 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1855 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1856 Minimierungsalgorithmus von einem NEA zu einem DEA]
1857 \liLadePakete{typographie}

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


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


1858 \def\liFussnote#1{$x_{\text{\scriptsize #1}}$}

1859 \def\li@fussnote@text#1#2{
1860   \liFussnote{\scriptsize #1}
1861   \quad
1862   {\footnotesize #2}
1863 }

1864 \def\liFussnoteEinsText{
1865   \li@fussnote@text{1}
1866   {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1867 }

1868 \def\liFussnoteZweiText{
1869   \li@fussnote@text{2}
1870   {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1871 }

1872 \def\liFussnoteDreiText{
1873   \li@fussnote@text{3}

```

```

1874 {In weiteren Iterationen markierte Zustände.}
1875 }

\liFussnoteVierText
1876 \def\liFussnoteVierText{
1877   \li@fussnote@text{4}
1878   {...}
1879 }

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



1880 \def\liFussnoten{
1881   \bigskip
1882
1883   \noindent
1884   \liFussnoteEinsText
1885
1886   \noindent
1887   \liFussnoteZweiText
1888
1889   \noindent
1890   \liFussnoteDreiText
1891
1892   \noindent
1893   \liFussnoteVierText
1894 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1895 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1896 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1897 \def\liZustandsPaar#1#2{
1898   $(
1899     \liZustandsPaarVariablenName_#1,
1900     \liZustandsPaarVariablenName_#2
1901   )$
1902 }

liUebergangsTabelle
1903 \renewcommand{\arraystretch}{1.4}
1904 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1905   \liPseudoUeberschrift{Übergangstabelle}
1906   \begin{center}
1907     \begin{tabular}{r|l|l}
1908       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
1909     \end{tabular}
1910   \end{center}
1911 }
1912 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1913 \ExplSyntaxOn
1914 \def\liUeberschriftDreiecksTabelle{
1915   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1916 }

```

\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. —

```

1917 \def\liMinimierungErklaerung{
1918   %\footcite[Seite~19]{koenig}
1919   \liParagraphMitLinien{
1920     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1921     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1922     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1923     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1924      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1925     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1926     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1927     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1928     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1929   }
1930 }
1931 \ExplSyntaxOff
1932

```

2.27 normalformen.sty

```

1933 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1934 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
1935 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1936 Attributhülle]
1937 \liLadePakete{mathe}
1938 \directlua{
1939   helfer = require('lehramt-informatik-helfer')
1940   normalformen = require('lehramt-informatik-normalformen')
1941 }
```

Let-Abkürzungen

```

\let\ah=\liAttributHuelle
\let\ahL=\liAttributHuelleLinksReduktion
\let\ahl=\liAttributHuelleLinksReduktionInline
\let\ahr=\liAttributHuelleLinksReduktionInline
\let\fa=\liFunktionaleAbhaengigkeit
\let\FA=\liFunktionaleAbhaengigkeiten
\let\m=\liAttributMenge
\let\r=\liRelation
\let\u=\underline
```

```

1942 \def\liTeilen#1{
1943   \directlua{tex.print(normalformen.teilen('#1'))}
1944 }
```

\liAttributHuelle Let-Abkürzung: $\text{AttrHülle}(F, \{A, B\})$ Regulärer Ausdruck zum Konvertieren $\text{AttrHülle}((.*) \setminus \setminus \text{ah}\{ \$1 \})$

```

1945 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
1946 \def\liAttributHuelle#1{
1947   \ifmmode
1948     \liAttributHuelleOhneMathe{\#1}
1949   \else
1950     $\liAttributHuelleOhneMathe{\#1}$
1951   \fi
1952 }
```

\liAttributMenge Let-Abkürzung: liAHuelle

```

1953 \def\liAttributMenge#1{\{ \textit{\#1} \}}
```

liAHuelle

```

1954 \NewDocumentEnvironment{ liAHuelle } { +b } {
1955   \begin{group}
1956   \footnotesize
1957   \begin{multline*}
1958     \#1
1959   \end{multline*}
1960   \endgroup
1961 } { }
```

AttributHuelleLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline

Let-Abkürzung: $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$

```

1962 \def\liAttributHuelleLinksReduktion#1#2#3{
1963   \shoveleft{
1964     \liAttributHuelleOhneMathe{FA,
1965       \liAttributMenge{\#1 \string\ #2}} =
1966     } \setminus
1967   \shoveright{
1968     \liAttributMenge{\#3}
```

```

1969 } \\\
1970 }

utHuelleLinksReduktionInline Let-Abkürzung: \let\ahL=\liAttributHuelleLinksReduktionInline
\ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
\ahL{A, E}{E}{A, E, F, B, \textbf{D}}: AttrHülle( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
1971 \def\liAttributHuelleLinksReduktionInline#1#2#3{
1972   $\liAttributHuelleOhneMathe{F,
1973     \liAttributMenge{#1 \string\ #2}} =
1974     \liAttributMenge{#3}$
1975 }

utHuelleLinksReduktionInline Let-Abkürzung: \let\ahr=\liAttributHuelleLinksReduktionInline
\ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
1976 \def\liAttributHuelleRechtsReduktionInline#1#2#3#4{
1977   $\liAttributHuelleOhneMathe{
1978     F \string\
1979     \liFunktionaleAbhaengigkeit{#1} \cup \liFunktionaleAbhaengigkeit{#2},
1980     \liAttributMenge{#3}
1981   } =
1982     \liAttributMenge{#4}$
1983 }

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
\liFunktionaleAbhaengigkeit{A, B -> C, D} Regulärer Ausdruck zum Konvertieren:
\$(.*?) \rightarrow (.*)\fa{$1 -> $2}
1984 \def\liFunktionaleAbhaengigkeit#1{%
1985   \directlua{normalformen.drucke_funk_abhaengigkeit('#1')}%
1986 }

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

Let-Abkürzung: \let\FA=\liFunktionaleAbhaengigkeiten
Regulärer Ausdruck zum Konvertieren: \item \$(.*) \rightarrow (.*)\fa{(.*) \rightarrow
$1 -> $2;
1987 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
1988   \par
1989   \noindent
1990   #1 $= \{$
1991   \par
1992   \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
1993   \par
1994   \noindent$\}$
1995 }

\liRelation Let-Abkürzung: \let\r=\liRelation
\r[R3]{\u{A}, B, C}:  $R_3(\underline{A}, B, C)$  Regulärer Ausdruck zum Konvertieren:
\$(R.*)\((.*)\)\fa{\liRelation[$1]{$2}
1996 \NewDocumentCommand {\liRelation} { O{R} m } {
1997   $\directlua{
1998     local name = helfer.konvertiere_tiefgestellt('#1')
1999     tex.print(name)
2000   }$(\textit{\, #2\,})
2001 }

2002

```

2.28 petri.sty

2003 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2004 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

Let-Abkürzungen

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2005 \RequirePackage{tikz}

2006 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2007 \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

```
2008 \def\liPetriSetzeSchluessel{%
2009   \def\TmpTransitionOne{}%
2010   \def\TmpTransitionTwo{}%
2011   \def\TmpTransitionThree{}%
2012   \def\TmpTransitionFour{}%
2013   \def\TmpTransitionFive{}%
2014   \def\TmpTransitionSix{}%
2015   \def\TmpTransitionSeven{}%
2016   \def\TmpTransitionEight{}%
2017   \def\TmpTransitionNine{}%
2018   \def\TmpTransitionTen{}%
2019   \pgfkeys{/petri/.cd,
2020     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2021     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2022     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2023     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
2024     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

2025     p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2026     p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2027     p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2028     p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2029     p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2030     t1/.store in=\TmpTransitionOne,t1/.default=activated,
2031     t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2032     t3/.store in=\TmpTransitionThree,t3/.default=activated,
2033     t4/.store in=\TmpTransitionFour,t4/.default=activated,
2034     t5/.store in=\TmpTransitionFive,t5/.default=activated,
2035     t6/.store in=\TmpTransitionSix,t6/.default=activated,
2036     t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2037     t8/.store in=\TmpTransitionEight,t8/.default=activated,
2038     t9/.store in=\TmpTransitionNine,t9/.default=activated,
2039     t10/.store in=\TmpTransitionTen,t10/.default=activated,
2040     scale/.store in=\TmpScale,scale/.default=0.5,
2041     x/.store in=\TmpX,x/.default=5,
2042     y/.store in=\TmpY,y/.default=5,
2043   }%
2044 }

2045 \tikzset{
2046   li petri/.style={
2047     activated/.style={
2048       very thick
2049     },
2050     inhibitor/.style={
2051       {Circle[open,length=2mm,fill=white]}-
2052     }
2053   }
2054 }

```

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

```

2055 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2056 \def\liPetriTransitionsName#1{
2057   \ifmmode
2058     \liPetriTransitionsNameOhneMathe{#1}
2059   \else
2060     $\liPetriTransitionsNameOhneMathe{#1}$
2061   \fi
2062 }

```

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

```

2063 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2064   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2065 }

```

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

```

2066 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2067 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2068 }

```

2.29 potenzmengen-konstruktion.sty

```
2069 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2070 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2071 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2072 \liLadePakete{formale-sprachen}
2073 \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}
    }
  }
}
```

```
2074 \def\liZustandsMengenSammlung#1#2{
2075   \liZustandsnameGross{#1}
2076   {
2077     \footnotesize
2078     \liPotenzmenge{
2079       \str_case:nn {#1} {#2
2080       }
2081     }
2082 }
```

```
\liZustandsMengenSammlungNr
```

```
2083 \def\liZustandsMengenSammlungNr#1#2{
2084   \liZustandsnameGross{#1}
2085   {
```



```
2086     \footnotesize
2087     \liZustandsmengeNr{
2088         \str_case:nn {#1} #2
2089     }
2090 }
2091 }

2092 \ExplSyntaxOff
2093
```

2.30 pseudo.sty

2094 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2095 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
 2096 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> <div style="margin-left: 20px;"> <p>wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;</p> <p>entferne die Kante e aus L;</p> <p>if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then</p> <div style="margin-left: 20px;"> <p>$E' \leftarrow E' \cup \{e\}$;</p> </div> <p>end</p> </div> <p>end</p> <p>Result: $M = (V, E')$ ist ein minimaler Spannbaum von G.</p>

2097 \RequirePackage[german,boxruled]{algorithm2e}

2098

2.31 pumping-lemma.sty

2099 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2100 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 2101 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2102 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2103 \def\liPumpingRegulaer{%
2104   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2105   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2106    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2107   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2108
2109   \begin{enumerate}
2110     \item  $|v| \geq 1$ 
2111     (Das Wort  $v$  ist nicht leer.)
2112
2113     \item  $|uv| \leq j$ 
2114     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2115
2116     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
2117     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2118     Sprache  $L$ )
2119   \end{enumerate}
2120
2121   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2122   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2123 }
```

\liPumpingKontextfrei

```
2124 \def\liPumpingKontextfrei{%
2125   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2126   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
2127    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2128
2129   \begin{enumerate}
2130     \item  $|vx| \geq 1$ 
2131     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2132
2133     \item  $|vwx| \leq j$ 
2134     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2135
2136     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy \in L$  (Für jede
2137     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2138     Sprache  $L$ )
2139   \end{enumerate}
2140 }
```

2141

2.32 quicksort.sty

```

2142 % https://tex.stackexchange.com/a/142634
2143 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2144 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2145
2146 %-----
2147 % USAGE:
2148 % \QSinitialize{comma, separated, numerical, values}
2149 % \loop
2150 % \QSpivotStep
2151 % \ifnum\value{pivotcount}>0
2152 %   \QSSortStep
2153 % \repeat
2154 %-----
2155
2156 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2157 \RequirePackage{xintfrac, xinttools}
2158
2159 \RequirePackage{tikz}
2160
2161 %-----
2162 % FIRST PART: TikZ styles and macros for the actual drawing
2163 \newcounter{cellcount}% used for coordinates of the node
2164 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2165
2166 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2167 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2168 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2169 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2170 % by police of LaTeX good conduct ? )
2171 \tikzset{ll/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2172          oo/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2173          rr/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2174 % this is the "b" style as used in the image below
2175          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2176 % nicer:
2177          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2178          gg/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2179
2180 % NOTE the b style was originally the same as the r(aised) style apart from
2181 % not being raised, but I find it nicer with a somewhat different
2182 % specification. I have not updated the images though.
2183
2184 % How the nodes are drawn depending on whether on the left of the pivot value
2185 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2186
2187 \def\DecoLEFT #1{%
2188   \xintFor* ##1 in {#1} \do
2189     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2190 }
2191
2192 \def\DecoINERT #1{%
2193   \xintFor* ##1 in {#1} \do
2194     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2195 }
2196
2197 \def\DecoRIGHT #1{%
2198   \xintFor* ##1 in {#1} \do
2199     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2200 }
2201
2202 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2203   \xintFor* ##1 in {#1} \do

```

```

2204     {\stepcounter{cellcount}}%
2205     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2206 }
2207
2208 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2209     \xintFor* ##1 in {#1} \do
2210     {\stepcounter{cellcount}%
2211     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2212 }
2213
2214 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2215     \xintFor* ##1 in {#1} \do
2216     {\stepcounter{cellcount}%
2217     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2218 }
2219
2220 %-----
2221 % SECOND PART: the actual sorting routines.
2222
2223 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2224 \def\QS@sort@b #1{\ifcase #1
2225     \expandafter\QS@sort@empty
2226     \or\expandafter\QS@sort@single
2227     \else\expandafter\QS@sort@c
2228     \fi
2229 }%
2230 \def\QS@sort@empty #1{}
2231 \def\QS@sort@single #1{\QSIr {#1}}
2232
2233 % This step is to pick the last as pivot.
2234 \def\QS@sort@c #1%
2235     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2236
2237 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2238 % The trick with \xintApplyUnbraced is that for example when selecting
2239 % the elements smaller than pivot, if we had been using \xintApply we
2240 % would have had at the minimum an empty brace pair. Thus we use the
2241 % "unbraced" variant, but then the \QS@select@smaller has added in
2242 % anticipation a level of braces.
2243 \def\QS@sort@d #1#2{%
2244     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}%
2245     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}%
2246     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}%
2247 }%
2248 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2249 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2250 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2251
2252 %
2253 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2254 %
2255 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2256 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2257 % latter must handle correctly an empty argument.
2258
2259 %-----
2260 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2261
2262 % This draws all with suitable highlighting for the newly chosen pivots
2263 % (which will be shown raised)

```

```

2264 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2265             \let\QSIr\DecoINERT
2266             \let\QSIrr\DecoINERT
2267             \let\QSRr\DecoRIGHTwithPivot
2268 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2269             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2270             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2271 }
2272
2273 % This sorts and then draws, showing where the pivot chosen in the previous
2274 % step go. Next time they will have become "inert". If pivotcount is still at
2275 % zero on exit from \QSpivotStep, then this is the signal to stop before
2276 % executing \QSsortStep.
2277 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2278             \def\QSRr {\noexpand\QS@sort@a}%
2279             \def\QSIr {\noexpand\QSIrr}%
2280             \let\QSIrr\relax
2281             \edef\QS@list{\QS@list}%
2282             \let\QSLr\relax
2283             \let\QSRr\relax
2284             \let\QSIr\relax
2285             \edef\QS@list{\QS@list}%
2286             \let\QSLr\DecoLEFT
2287             \let\QSIr\DecoINERTwithPivot
2288             \let\QSIrr\DecoINERT
2289             \let\QSRr\DecoRIGHT
2290 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2291             \setcounter{cellcount}{0}%
2292             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2293 }
2294
2295 \def\QSinitialize #1{%
2296     % first, we convert the comma separated values into a list of braced items
2297     % we use an \edef, and anyhow many \edef's will be used later
2298     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2299     \let\QSRr\DecoRIGHT
2300     % The \QSRr marker mutated to draw the last element as
2301     % pivot and the earlier ones with the suitable style.
2302     %
2303     % The list of marked braced items \QS@list is used both for drawing
2304     % (as here) and for doing the exchange of elements during sort.
2305     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2306             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2307 }
2308

```

2.33 relationale-algebra.sty

2309 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2310 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]

2311 \RequirePackage{amsmath}

2312 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

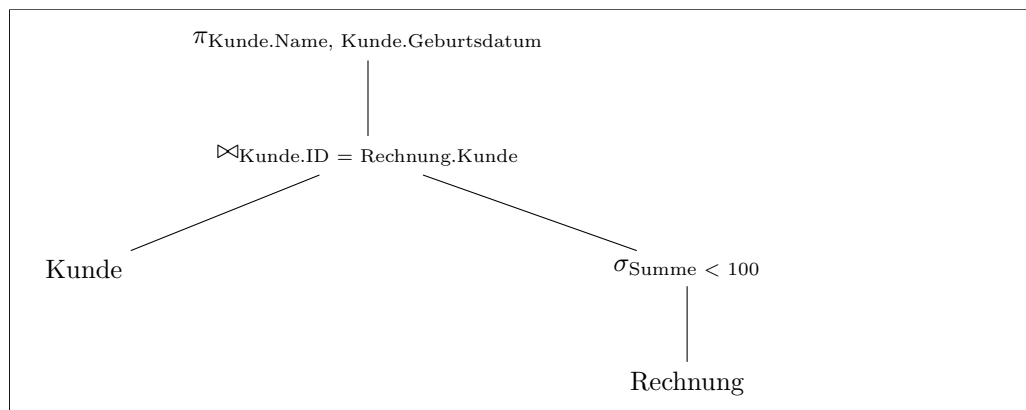
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

  \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}
```



2313 \RequirePackage{tikz}

2314 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

```
2315 \def\o@join{\setbox0=\hbox{\bowtie}}%
2316 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2317 }
```

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

```
2318 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

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

```
2319 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

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

```
2320 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

2321

2.34 rmodell.sty

```

2322 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2323 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01
2324 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2325 Datenbanken.]
2326 \RequirePackage{soul}

```

Let-Abkürzungen

```

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

```

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

```
2327 \def\liPrimaer#1{\ul{#1}}
```

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

```
2328 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```

2329 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2330 \ExplSyntaxOn
2331 \NewDocumentEnvironment { liRmodell }
2332 { +b }
2333 {
2334   \medskip
2335   {
2336     \linespread{2}
2337     \setlength{\parindent}{0pt}
2338     \li@Rmodell@Schrift#1
2339   }
2340   \medskip
2341 } {}
2342 \ExplSyntaxOff

```

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

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

```

2343 \def\liRelationMenge#1#2{
2344 \noindent
2345 #1 : \[ #2 ]\}
2346 \par
2347 }

```

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

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

```
2348 \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}

```

```
2349 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2350
```


2.35 sortieren.sty

```
2351 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2352 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2353 für händisches Sortieren bzw. einen Schreibtischlauf setzen]
```

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



```
2354 \RequirePackage{tikz}
2355 \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.

```
2356 \def\liVertauschen#1{
2357   \directlua{
2358     local sortieren = require('lehramt-informatik-sortieren')
2359     sortieren('#1')
2360   }
2361 }
```

`\liSortierPfeil`

```
2362 \def\liSortierPfeil#1#2{
2363   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2364 }
```

`\liSortierPfeilUnten`

```
2365 \def\liSortierPfeilUnten#1#2{
2366   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2367 }
```

`\liSortierMarkierung`

```
2368 \def\liSortierMarkierung#1#2{\node[
2369   draw,
2370   very thick,
2371   fit=(reihe.#1) (reihe.#2),
2372   inner sep=0pt
2373 ] {}
2374 }

2375 \tikzset{
2376   li sortierung zahlenreihe/.style={
2377     draw,
2378     thin,
2379     font=\large,
2380     rectangle split horizontal,
2381     rectangle split,
2382   }
2383 }
```

```

2384 % https://tex.stackexchange.com/a/140895
2385 \RequirePackage{forest,xstring}
2386 \usetikzlibrary{calc}
2387
2388 \makeatletter
2389 \pgfmathdeclarefunction{strrepeat}{2}{%
2390   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2391   \let\pgfmathresult\pgfutil@empty
2392   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2393     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2394     \advance\pgfmath@count-1\relax
2395   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2396 \makeatother
2397
2398 \def\myNodes{}
2399
2400 \ExplSyntaxOn
2401 \newcommand*\sortList[1]{%
2402   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2403 \ExplSyntaxOff
2404
2405 \forestset{
2406   sort/.code={%
2407     \pgfmathparse{level()}>\forestSortLevel}%
2408     \ifnum\pgfmathresult=0
2409       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2410       \sortList\myList
2411       \StrSubstitute{\myList}{,}{ }{\myList}%
2412       \pgfmathparse{strrepeat("1",level())}%
2413       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2414         (m\forestov{name}) {\myList}}%
2415       \pgfmathparse{level()}==\forestSortLevel}%
2416       \ifnum\pgfmathresult=1
2417         \forestOget{\forestov{@first}}{name}\forestFirst
2418         \forestOget{\forestov{@last}}{name}\forestLast
2419         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2420           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2421       \fi
2422       \ifnum\forestov{@parent}=0\else
2423         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2424       \fi
2425       \gappto\myNodes{;}%
2426     \fi}}
2427
2428 \forestset{sort level/.code=%
2429   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2430   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2431

```

2.36 spalten.sty

```
2432 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2433 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2434 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2435 realisiert werden kann.]
2436 \RequirePackage{multicol}
```

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

```
2437 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2438
```

2.37 struktogramm.sty

```
2439 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2440 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2441 Paket struktex zum Setzen von Struktogrammen]
2442 \RequirePackage{struktex}
2443
```

2.38 syntax.sty

```
2444 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2445 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2446 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2447 \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
```

```
2448 \ExplSyntaxOn
2449 \directlua{
2450   syntax = require('lehramt-informatik-syntax')
2451   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2452   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2453   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2454   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2455   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2456   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2457 }
2458 \RequirePackage{hyperref}
2459 \RequirePackage{minted}
2460 % pygmentize -L styles
2461 \usemintedstyle{colorful}
2462 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2463 %\AfterEndEnvironment{minted}{\end{mdframed}}
2464 %\setminted{breaklines=true,linenos}
2465 \setminted{
2466   breaklines=true,
2467   linenos,
2468   fontsize=\footnotesize,
2469 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2470 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2471 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2472 \def\li@GithubLink#1#2{
2473   \begin{flushright}
2474     \tiny
2475     Code-Beispiel~auf~Github~ansehen:~
2476     \href{#1}{\nolinkurl{#2}}
2477   \end{flushright}
2478 }
```

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

```
2479 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2480   \inputminted[#1]{java}{
2481     \directlua{
2482       syntax.drucke_absoluten_pfad('#2', false)
2483     }
2484   }
2485   \li@GithubLink
```

```

2486     {\directlua{syntax.drucke_github_url('#2', false)}}
2487     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2488 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2489 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2490   \inputminted[#1]{java}{
2491     \directlua{
2492       syntax.drucke_absoluten_pfad('#2', true)
2493     }
2494   }
2495   \li@GithubLink
2496   {\directlua{syntax.drucke_github_url('#2', true)}}
2497   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2498 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2499 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2500   \inputminted[#1]{java}{
2501     \directlua{
2502       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2503     }
2504   }
2505   \li@GithubLink
2506   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2507   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2508   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2509 }

\liAssemblerCode
2510 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2511 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2512   \inputminted{asm}{#1}
2513 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2514 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2515   \inputminted{componentpascal}{#1}
2516 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2517 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2518 \NewDocumentCommand{\liHaskellDatei}{ m }{
2519   \inputminted{haskell}{#1}
2520 }

2521 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2522 \def\liSqlCode#1{\mintinline{sql}|#1|}
2523

```

2.39 syntaxbaum.sty

```
2524 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2525 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2526 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2527 \RequirePackage{tikz-qtrees}
2528
2529 \tikzset{li parsetree/.style={
2530     every internal node/.style={
2531         draw,circle
2532     },
2533     every leaf node/.style={
2534         draw,rectangle
2535     },
2536 }
2537 }
2538
```

2.40 synthese-algorithmus.sty

```
2539 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2540 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2541 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2542 Relation in die 3. Normalform]

2543 \liLadePakete{normalformen,mathe,typographie}
2544 \ExplSyntaxOn
```

Let-Abkürzungen

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

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

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

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

1. Kanonische Überdeckung

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

(a) Linksreduktion

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

(b) Rechtsreduktion

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

(c) Löschen leerer Klauseln

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

(d) Vereinigung

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

2. Relationsschemata formen

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

3. Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata \mathcal{R}_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 R_α , die in einem anderen Relationenschema $R_{\alpha'}$ enthalten sind, d. h. $R_\alpha \subseteq R_{\alpha'}$.* —

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

```
2545 \def\liSyntheseUeberschrift#1{
2546   {
2547     \bfseries
2548     \sffamily
2549     \str_case:nn {#1} {
2550       {1} {Kanonische-Überdeckung}
2551       {1-1} {Linksreduktion}
2552       {1-2} {Rechtsreduktion}
2553       {1-3} {Löschen-leerer-Klauseln}
2554       {1-4} {Vereinigung}
2555       {2} {Relationsschemata-formen}
2556       {3} {Schlüssel-hinzufügen}
2557       {4} {Entfernung-überflüssiger-Teilschemata}
2558     }
2559   }
2560 }
```

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

```
2561 \def\li@synthese@erklaerung@texte#1{
2562   \str_case:nn {#1} {
2563     {1} {
2564       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2565       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2566       Schritten-erreicht-werden.
2567     }
2568     {1-1} {
2569       Führe-für-jede-funktionale-Anhängigkeit-
2570       $\alpha\rightarrow\beta$-in-F$-die-Linksreduktion-durch,-
2571       überprüfe-also-für-alle-
2572       $A$-in-$\alpha$,~ob-$A$-überflüssig-ist,~d.h.-ob-
2573       $\beta\subseteqq\liAttributHuelle{F,\alpha\---A}$.
2574     }
2575     {1-2} {
2576       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2577       alle-$B$-in-$\beta$,~ob-$B$-in-$\liAttributHuelle{F---(\alpha\rightarrow\beta)\cup(\alpha\rightarrow\beta\---B)}$,~
2578       $\alpha$-gilt.-In-diesem-Fall-ist-B-auf-der-rechten-Seite-
2579       überflüssig-und-kann-eliminiert-werden,~dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow(\beta\---B)$-
2580       ersetzt.
2581     }
2582     {1-3} {
2583       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2584       entstanden-sind.
2585     }
2586     {1-4} {
2587       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2588       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.
2589     }
2590     {2} {
2591       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-in-F$\sb{c}$-ein-Relationenschema-$\mathcal{R}\sb{\alpha}$-
2592     }
2593   }
```

```

2600      :=~\alpha~\cup~\beta$.
2601    }
2602    {3} {
2603      Falls-eines-der-in-Schritt-2.-erzeugten-Schemata- $R_{\alpha}$ -
2604      einen-Schlüsselkandidaten-von- $R$ -bezüglich- $F_{\mathcal{C}}$ -
2605      enthält,~sind-wir-fertig,~sonst-wähle-einen-Schlüsselkandidaten-
2606       $K \subseteq R$ -aus-und-definiere-folgendes-
2607      zusätzliche-Schema:~ $R_{\mathcal{C}} \setminus R_K := \emptyset$ -
2608      und- $F_{\mathcal{C}} \setminus R_K := \emptyset$ 
2609    }
2610    {4} {
2611      Eliminiere-diejenigen-Schemata- $R_{\alpha}$ ,~die-in-einem-
2612      anderen-Relationenschema- $R_{\alpha'}$ -enthalten-sind,~d.h.~
2613       $R_{\alpha} \subseteq R_{\alpha'}$ .
2614    }
2615  }
2616 }
2617 \def\liSyntheseErklaerung#1{
2618   {
2619     \itshape
2620     \footnotesize
2621     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2622   }
2623 }

```

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

```

2624 \def\liSyntheseUeberErklaerung#1{
2625   \liSyntheseUeberschrift{#1}\par
2626   \liSyntheseErklaerung{#1}
2627 }

2628 \ExplSyntaxOff
2629

```

2.41 tabelle.sty

2630 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2631 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2632 \RequirePackage{tabularx}

2633

2.42 typographie.sty

```
2634 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2635 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2636 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2637 formatierung.sty definiert.]
```

```
2638 \ExplSyntaxOn
```

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

```
2639 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2640 \let\liErledigt=\faCheckSquareO
```

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

```
2641 \def\liNichtsZuTun{${\emptyset}$~Nichts~zu~tun}
```

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

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

```
2642 \def\liParagraphMitLinien#1{
```

```
2643 \noindent
```

```
2644 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2645 \enspace
```

```
2646 #1
```

```
2647 \enspace
```

```
2648 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2649 \par
```

```
2650 \medskip
```

```
2651 }
```

```
2652 \ExplSyntaxOff
```

```
2653
```

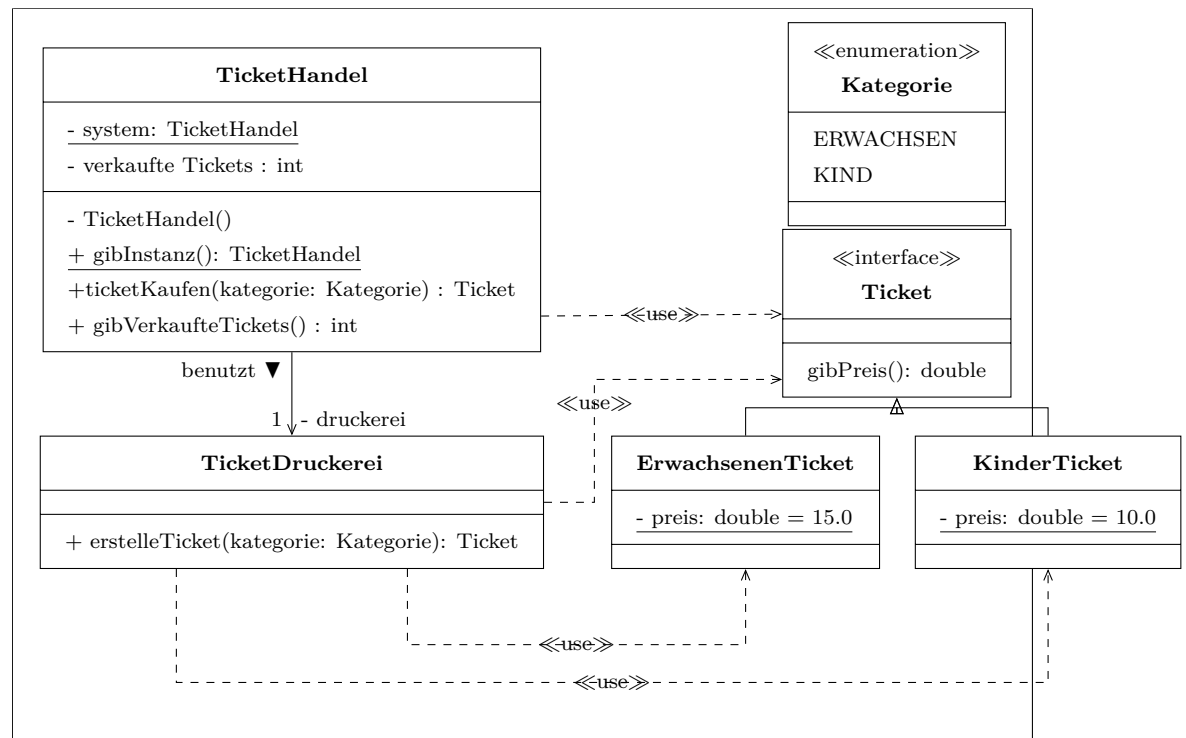
2.43 uml.sty

```

2654 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2655 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2656 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2657 Erweiterung bereitstellt]

2658 \RequirePackage{tikz-uml}
2659 \RequirePackage{tikz-uml-activity}
2660 % Not compatible with wasysym
2661 %\RequirePackage{mathabx}
2662 \RequirePackage{wasysym}
2663 \usetikzlibrary{positioning}

2664 \tikzumlset{
2665   fill class=white!0,
2666   font=\footnotesize,
2667   fill object=white!0,
2668   fill note=white!0,
2669   fill state=white!0,
2670   % Use case
2671   fill usecase=white!0,
2672   fill system=white!0,
2673 }
```



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

```

2674 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2675   \def\@liDirLeft{}
2676   \def\@liDirRight{ \RIGHTarrow}
2677   \pgfkeys{/lese/dir/.is choice}
2678   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2679   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2680   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2681   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2682
2683   \def\@liPos{above}
2684   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2685

```

```

2686 \def\@liDistance{0cm}
2687 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2688
2689 \pgfkeys{/lese/.cd,#1}
2690
2691 \node[\@liPos = \@liDistance of #2-middle] {
2692   \@liDirLeft{\footnotesize#2}\@liDirRight
2693 };
2694 }
2695

```

2.44 vollstaendige-induktion.sty

2696 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2697 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2698 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2699 Überschriften für die einzelnen Schritte]

Let-Abkürzungen

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

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

Lade häufig benötigte Pakete

```
2700 \RequirePackage{lehramt-informatik-typographie}
2701 \RequirePackage{lehramt-informatik-mathe}
2702 \RequirePackage{lehramt-informatik-syntax}
2703 \ExplSyntaxOn
```

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

```
2704 \def\liInduktionMarkierung#1{\textcolor{violet}{#1}}
```

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

Let-Abkürzung: \let\e=\liInduktionErklaerung

2705 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}

\liInduktionAnfang

2706 \def\liInduktionAnfang{

2707 \liPseudoUeberschrift{Induktionsanfang}

2708

2709 % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion

2710 \liParagraphMitLinien{

2711 Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.

2712 }

2713 }

\liInduktionVoraussetzung

2714 \def\liInduktionVoraussetzung{

2715 \liPseudoUeberschrift{Induktionsvoraussetzung}

2716

2717 % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion

2718 \liParagraphMitLinien{

2719 Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.

2720 }

2721 }

\liInduktionSchritt

2722 \def\liInduktionSchritt{

2723 \liPseudoUeberschrift{Induktionsschritt}

2724

2725 % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion

2726 \liParagraphMitLinien{

2727 Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~

2728 auch~ $A(n=k+1)$ ~wahr~sein~muss.

2729 }

2730 }

2731 \ExplSyntaxOff

2732

2.45 wasserfall.sty

```
2733 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2734 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2735 \RequirePackage{tikz}
2736 \tikzset{wasserfall/.style={
2737   >=stealth,
2738   node distance = 2mm and -8mm,
2739   start chain = A going below right,
2740   every node/.style = {
2741     draw,
2742     text width=24mm,
2743     minimum height=12mm,
2744     align=center,
2745     inner sep=1mm,
2746     fill=white,
2747     drop shadow={fill=black},
2748     on chain=A
2749   },
2750 }}
2751 \usetikzlibrary{chains,positioning,shadows}
2752
```

2.46 wpkalkuel.sty

2753 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2754 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2755 \RequirePackage{amsmath}

2756 \ExplSyntaxOn

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

2757 \def\liWpKalkuelOhneMathe#1#2{

2758 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2759 }

2760 \def\liWpKalkuel#1#2{

2761 \ifmmode

2762 \liWpKalkuelOhneMathe{#1}{#2}

2763 \else

2764 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2765 \fi

2766 }

\MatheEnv

2767 \def\MatheEnv#1{

2768 \medskip

2769

2770 \hspace{1em}#1

2771

2772 \medskip

2773 }

\Mathe

2774 \def\Mathe#1{

2775 \MatheEnv{#1\$}

2776 }

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

2777 \def\liWpEquivalent#1{

2778 \MatheEnv{\$\equiv\$\hspace{1em}#1\$}

2779 }

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

2780 \newlength{@Skip@Erklaerung@Reset}

2781 \def\liWpErklaerung#1{

2782 \setlength{@Skip@Erklaerung@Reset}{\leftskip}

2783 \setlength{\leftskip}{0.5cm}

2784

2785 \par

2786 \noindent

2787 {

2788 \scriptsize

2789 #1

2790 }

2791 \par

2792

2793 \setlength{\leftskip}{@Skip@Erklaerung@Reset}

2794 }

\liWpErklaerungVerzweigung

```

2795 \def\liWpErklaerungVerzweigung{
2796   $\liWpKalkuelOhneMathe{if~\{-b~\}-then~\{-a1~\}-else~\{-a2~\}}{Q}
2797   \equiv
2798   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2799   \lor
2800   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2801 }

2802 \ExplSyntaxOff

2803

```

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	\allsectionsfont ... 1223	\bigskip 48, 364,
\# 109	\Alph 1237	598, 603, 1553, 1881
\, 331, 386,	\alph 1237, 1238	\bool 309, 332
1174, 1176, 1677,	\alpha 2570, 2572, 2573,	\bowtie 2315, 2318, 2319, 2320
1678, 1679, 2000, 2470	2576, 2578, 2579,	\Box 159
\@Skip@Erklaerung@Reset	2580, 2581, 2582,	\boxtimes 463
... 2780, 2782, 2793	2586, 2592, 2593,	
\@afterheading 1644	2598, 2599, 2600,	
\@afterindentfalse . 1644	2603, 2611, 2612, 2613	C
\@liDirLeft 2675, 2680, 2692	\arabic 1237, 2189, 2194,	\c 1320, 1321
\@liDirRight 2676, 2678,	2199, 2205, 2211, 2217	\cdot 1721, 1774, 1785
2679, 2680, 2681, 2692	\arraystretch 1903	\centerline 1351, 2268, 2290, 2305
\@liDistance 2686, 2687, 2691		\chapter 1227, 1228
\@liPos .. 2683, 2684, 2691	B	\char 1551
\ .. 596, 619, 620, 623,	\BeforeBeginEnvironment	\clearpage 1545
624, 627, 628, 720, 2462	\cline 596
721, 722, 829, 858,	\begin 617, 668,	\clist 226, 270,
860, 886, 895, 940,	683, 718, 742, 789,	271, 284, 288, 2402
982, 983, 984, 989,	821, 836, 856, 866,	\columnbreak 2437
990, 991, 1011,	882, 902, 934, 951,	\cs 287, 306, 330,
1551, 1908, 1966, 1969	980, 1005, 1026,	331, 368, 380, 1627
\{ 207, 1109, 1119,	1041, 1153, 1157,	\csname 1305, 1308
1131, 1132, 1137,	1243, 1318, 1346,	\cup 1132,
1151, 1173, 1392,	1355, 1362, 1470,	1979, 2579, 2593, 2600
1953, 1990, 2345, 2796	1543, 1595, 1600,	
\} 207, 1109, 1119,	1608, 1633, 1638,	D
1131, 1132, 1139,	1649, 1659, 1663,	\DeclareMathSymbol ..
1157, 1177, 1393,	1746, 1750, 1766, 1851, 1852
1953, 1994, 2345, 2796	1789, 1812, 1827,	\DecoINERT 2192, 2265, 2266, 2288
_ 27, 35, 44, 46, 287, 306,	1906, 1907, 1957,	\DecoINERTwithPivot .
330, 331, 345, 346,	2109, 2129, 2270, 2208, 2287
352, 355, 358, 368, 380	2292, 2306, 2462, 2473	\DecoLEFT 2187, 2286
	\begingroup 1544, 1955, 2390	\DecoLEFTwithPivot ..
	\beschriftung 1575, 1579, 2202, 2264
\sq 1965, 1973, 1978	1583, 1587, 1591, 1593	\DecoRIGHT 2197, 2289, 2299
A	\beta 2570,	\DecoRIGHTwithPivot .
\addbibresource 1521, 1522, 1523,	2573, 2577, 2578, 2214, 2267
1524, 1525, 1526,	2579, 2582, 2592,	\definecolor 1225
1527, 1528, 1529, 1530	2593, 2594, 2599, 2600	\delta 65, 107, 165, 207, 1123
\advance 2394	\bf 2167, 2168, 2169	\dh 1679, 2581
\AfterEndEnvironment 2463	\bfseries .. 475, 1227,	\directlua 58, 137, 195, 200,
	1229, 2167, 2173,	
	2175, 2177, 2178, 2547	

1103, 1118, 1138, 1146, 1154, 1160, 1938, 1943, 1985, 1992, 1997, 2357, 2449, 2481, 2486, 2487, 2491, 2496, 2497, 2501, 2507, 2508	liRelationenSchemaFormat 2349 liRmodell 2329 liUebergangsTabelle 1903 \equiv 2778, 2797 \erzeuge@tiefgestellt ... 1118, 1119, 1123	\forestLast ... 2418, 2420 \forestOget ... 2417, 2418 \forestOnes 2430 \forestOv 2419, 2420, 2423 \forestov . 2409, 2413, 2414, 2417, 2418, 2419, 2420, 2422, 2423 \forestset 2405, 2428 \forestSortLevel 2407, 2415, 2429, 2430 \frac 1723, 1754, 1785, 1800 \fullouterjoin 2320
\do 2188, 2193, 2198, 2203, 2209, 2215 \dots 506, 510, 1392, 2116, 2592, 2593 \DOWNarrow 2679 \draw 1325, 1328, 1331, 2064, 2363, 2366	\expandafter 1305, 2223, 2225, 2226, 2227, 2235, 2393 \ExplSyntaxOff 50, 92, 134, 139, 192, 197, 202, 393, 528, 550, 565, 1097, 1179, 1214, 1490, 1505, 1669, 1744, 1931, 2092, 2342, 2403, 2521, 2628, 2652, 2731, 2802 \ExplSyntaxOn 22, 61, 102, 135, 160, 193, 198, 223, 470, 534, 551, 1070, 1169, 1191, 1483, 1495, 1570, 1683, 1913, 2073, 2330, 2400, 2448, 2544, 2638, 2703, 2756	G \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 \Gamma . 106, 164, 207, 1132 \gappto 2425 \geq 1399, 2105, 2110, 2126, 2130
E \edef 1316, 2281, 2285, 2297, 2298 \else 570, 578, 586, 1113, 1127, 1165, 1591, 1694, 1704, 1714, 1728, 1949, 2059, 2227, 2420, 2422, 2763 \emph 1073, 1373, 1402, 1404, 1550 \empty 1591 \emptyset 1895, 2587, 2608, 2641 \end 656, 679, 704, 739, 773, 805, 833, 849, 863, 871, 898, 927, 948, 973, 998, 1018, 1038, 1056, 1155, 1157, 1245, 1334, 1358, 1359, 1366, 1475, 1548, 1597, 1605, 1624, 1640, 1641, 1655, 1666, 1667, 1748, 1761, 1786, 1809, 1824, 1834, 1910, 1911, 1959, 2119, 2139, 2270, 2292, 2306, 2463, 2477 \endcsname 1305, 1308 \endgroup 1547, 1960, 2395 \enspace 2645, 2647 environments: liAdditum 1598 liAHuelle 1954 liAntwort 1569 liDiagramm 1657 liEinbettung 1568 liExkurs 1606 liGraphenFormat . 1297 liKasten 1242 liKontrollflussgraph 1469 liLernkartei 1647 liProduktionsRegeln 1147 liProjektSprache 1567 liQuellen 1627	F \faCheckSquare0 2640 \faCircleThin 1089 \faGg 1083 \fancyfoot 1249, 1250, 1251, 1499, 1500, 1501, 1502 \fancyhead 1248, 1496, 1497, 1498 \faSquare0 1077 \fi 572, 580, 588, 1115, 1129, 1167, 1594, 1696, 1706, 1716, 1730, 1951, 2061, 2228, 2420, 2421, 2424, 2426, 2765 \fontspec 1223 \footcite 680, 702, 749, 772, 804, 899, 926, 972, 1371, 1374, 1381, 1386, 1391, 1395, 1401, 1406, 1510, 1762, 1763, 1918, 2122 \footnote 1671, 1675 \footnotesize 147, 341, 424, 520, 1094, 1566, 1619, 1634, 1650, 1862, 1956, 2077, 2086, 2329, 2468, 2620, 2666, 2692 \footrulewidth . 1254, 1504 \foreach . 1320, 1323, 1330 \forestFirst .. 2417, 2420	H \hbox 2315 \headrulewidth . 1253, 1503 \hfill 1770, 1776, 1781, 2648 \hinweis 1566 \hline 1908 \href 370, 1187, 1675, 1838, 2476 \hspace .. 2067, 2770, 2778 \ht 2316 I \i 1330, 1331 \ifcase 2224 \ifmode 568, 576, 584, 1111, 1125, 1163, 1692, 1702, 1712, 1726, 1947, 2057, 2761 \IfNoValueTF 1601, 1671, 1675 \ifnum 2151, 2392, 2408, 2416, 2422 \ifx 1591, 2420 \in 494, 600, 1399, 1736, 1739, 1742, 1768, 1774, 1779, 2105, 2116, 2126, 2136, 2570, 2572, 2578, 2599, 2719 \inhaltsverzeichnis 1542 \input . 4, 7, 10, 13, 16, 394 \inputminted 2480, 2490, 2500, 2512, 2515, 2519

<code>\int</code>	2402	<code>\LEFTarrow</code>	2680	<code>liAHuelle</code> (environment)	
<code>\item</code>	463,	<code>\leftarrow</code>	582		1954
	464, 685, 689, 694,	<code>\leftouterjoin</code>	2318	<code>\liAlphabet</code>	1131
	699, 743, 752, 757,	<code>\leftskip</code> 2782, 2783, 2793		<code>liAntwort</code> (environment)	
	765, 837, 842, 846,	<code>\LehramtInformatikAutorEmail</code>			1569
	867, 903, 908, 915,	1502	<code>\liAnweisung</code>	1477
	923, 952, 957, 961,	<code>\LehramtInformatikAutorName</code>		<code>\liAssemblerCode</code> ...	2510
	966, 1042, 1047,	1501	<code>\liAssemblerDatei</code> ..	2511
	1052, 1356, 1357,	<code>\LehramtInformatikGitBranch</code>		<code>\liAttribut</code>	2348
	1627, 1631, 1751,	373, 2456	<code>\liAttributHuelle</code> ...	
	1754, 1758, 1767,	<code>\LehramtInformatikGithubCodeRepo</code>		... 1945, 2573, 2578	
	1773, 1778, 1790,	2455	<code>\liAttributHuelleLinksReduktion</code>	
	1794, 1798, 1802,	<code>\LehramtInformatikGithubDomain</code>		1962
	1806, 1813, 1817,	2452	<code>\liAttributHuelleLinksReduktionInline</code>	
	1821, 2110, 2113,	<code>\LehramtInformatikGithubRawDomain</code>		1971, 1976
	2116, 2130, 2133, 2136	371, 2453	<code>\liAttributHuelleOhneMathe</code>	
<code>\itshape</code>	519, 2619	<code>\LehramtInformatikGithubTexRepo</code>		... 1945, 1948,	
		372, 2454	1950, 1964, 1972, 1977	
		<code>\LehramtInformatikRepository</code>		<code>\liAttributHuelleRechtsReduktionInline</code>	
		4,	1976
			7, 10, 13, 16, 1521,	<code>\liAttributMenge</code>	
			1522, 1523, 1524,	1953, 1965, 1968,	
			1525, 1526, 1527,	1973, 1974, 1980, 1982	
			1528, 1529, 1530, 2451	<code>\liAufgabe</code>	3
<code>\LehramtInformatikTitel</code>				<code>\liAufgabenTitel</code>	23
.....	1497			<code>\liAusdruck</code>	1170
<code>\leq</code>	1785, 2113, 2133			<code>\liAutomat</code>	61
<code>\let</code>	1121,			<code>\liAutomatenKante</code>	93
	1122, 1545, 2264,			<code>\liBandAlphabet</code>	1132
	2265, 2266, 2267,			<code>\liBedingung</code>	1478
	2280, 2282, 2283,			<code>\liBedingungDrei</code>	
	2284, 2286, 2287,			... 1741, 1781, 1821	
	2288, 2289, 2299,			<code>\liBedingungEins</code>	
	2391, 2429, 2430, 2640			... 1735, 1770, 1813	
<code>\li@chomsky@erklaerung@texte</code>				<code>\liBedingungFalsch</code> .	1480
.....	485, 521			<code>\liBedingungWahr</code> ...	1479
<code>\li@EntwurfsCode</code>				<code>\liBedingungZwei</code>	
	613, 659, 660, 661,			... 1738, 1776, 1817	
	707, 708, 709, 710,			<code>\liBeschriftung</code>	1558
	776, 777, 778, 779,			<code>\liChomskyErklaerung</code>	
	780, 781, 808, 809,			485, 526
	810, 811, 812, 813, 874			<code>\liChomskyUeberErklaerung</code>	
<code>\li@EntwurfsCodeAllgemein</code>				524
.....	612			<code>\liChomskyUeberschrift</code>	
<code>\li@fussnote@text</code> 1859,				473, 525
1865, 1869, 1873, 1877				<code>\liCpmEreignis</code>	534
<code>\li@GithubLink</code>				<code>\liCpmFruehsterI</code> ...	591
2472, 2485, 2495, 2506				<code>\liCpmSpaetesterI</code> ...	590
<code>\li@mget</code> . 1307, 1311, 1331				<code>\liCpmVon</code>	574
<code>\li@minc</code>	1310, 1332			<code>\liCpmVonOhneMathe</code> ..	
<code>\li@mset</code>	574, 577, 579
1304, 1312, 1321, 1324				<code>\liCpmVonZu</code>	566
<code>\li@numdiscs</code>				<code>\liCpmVonZuOhneMathe</code>	
... 1316, 1325, 1331				566, 569, 571
<code>\li@Rmodell@Schrift</code> .				<code>\liCpmVorgang</code>	551
... 2329, 2338, 2348				<code>\liCpmZu</code>	582
<code>\li@sequence</code> .. 1317, 1330				<code>\liCpmZuOhneMathe</code> ...	
<code>\li@synthese@erklaerung@text</code>				582, 585, 587
.....	2561, 2621			<code>\PiDiagramm</code> (environ-	
<code>\liAbleitung</code>	1146			ment)	1657
<code>liAdditum</code> (environment)				<code>liEinbettung</code> (environ-	
.....	1598			ment)	1568
<code>\left</code>	1686			<code>\liEntwurfs</code>	1021

\liEntwurfsAbstrakteFabrik	\liEntwurfsModellPraesentation	\liStellenDiagramm ..
..... 663 1004, 1022 135, 141
\liEntwurfsAbstrakteFabrikCode	\liEntwurfsZustand .. 1058	\liKontrollCode 1481
..... 658, 665	\liEntwurfsZustandAkteure	\liKontrollflussgraph
\liEntwurfsAbstrakteFabrikUml 1040, 1060	(environment) 1469
..... 616, 664	\liEntwurfsZustandUml	\liKontrollKnotenPfad
\liEntwurfsAdapter .. 712 1025, 1059 1483
\liEntwurfsAdapterAkteure	\liEpsilon 1117	\liKontrollTextzeileKnoten
..... 682, 714	\liErAttribute 1482, 1487
\liEntwurfsAdapterCode	... 1073, 1087, 1089	\liKurzeTabellenLinie 596
..... 706, 715	\liErDatenbankName .. 1092	\liLadeAllePakete ... 228
\liEntwurfsAdapterUml	\liErEntity 1071, 1075, 1077	\liLadePakete
..... 667, 713	\liErledigt 2640	. 54, 57, 224, 229,
\liEntwurfsBeobachter 783	\liErMpAttribute ... 1086	472, 533, 1341,
\liEntwurfsBeobachterAkteure	\liErMpEntity 1074	1857, 1937, 2072, 2543
..... 741, 785	\liErMpRelationship 1080	\liLatexCode 2471
\liEntwurfsBeobachterCode	\liErRelationship ...	\liLeereZelle 1895
..... 775, 786	... 1072, 1081, 1083	\liLernkartei (environ-
\liEntwurfsBeobachterUml	\liExamensAufgabe 6	ment) 1647
..... 717, 784	\liExamensAufgabeA ... 15	\liMasterExkurs 1826
\liEntwurfsDekorierer 815	\liExamensAufgabeTA .. 12	\liMasterFaelle 1765, 1833
\liEntwurfsDekoriererAkteure	\liExamensAufgabeTTA .. 9	\liMasterFallRechnung
..... 817	\liExkurs (environment) 1606 1811
\liEntwurfsDekoriererCode	\liFalsch 464	\liMasterVariablen ..
..... 807, 818	\liFlaci 1180 1745, 1828
\liEntwurfsDekoriererUml	\liFremd 2328	\liMasterVariablenDeklaration
..... 788, 816	\liFunktionaleAbhaengigkeit 1788
\liEntwurfsEinfacheFabrik 1979, 1984	\liMasterWolframLink 1836
..... 851	\liFunktionaleAbhaengigkeiten	\liMenge ... 71, 72, 74,
\liEntwurfsEinfacheFabrikAkteure 1987	113, 114, 115, 119,
..... 835, 853	\liFussnote ... 1858, 1860	171, 172, 173, 177,
\liEntwurfsEinfacheFabrikUml	\liFussnoteDreiText ..	1109, 1160, 1199, 1200
..... 820, 852 1872, 1890	\liMengeOhneMathe ...
\liEntwurfsEinzelstueck	\liFussnoteEinsText 1109, 1112, 1114
..... 876 1864, 1884	\liMinimierungErklaerung
\liEntwurfsEinzelstueckAkteure	\liFussnoteLink 1674 1917
..... 865, 878	\liFussnoten 1880	\liMinispracheDatei 2514
\liEntwurfsEinzelstueckCode	\liFussnoteUrl .. 1019, 1670	\linespread 2336
..... 873, 879	\liFussnoteVierText ..	\liNichtsZuTun 2641
\liEntwurfsEinzelstueckUml 1876, 1893	\liO 1708, 1736
..... 855, 877	\liFussnoteZweiText ..	\liOmega 1698, 1742
\liEntwurfsErbauer .. 929 1868, 1887	\liOmegaOhneMathe ...
\liEntwurfsErbauerAkteure	\liGrammatik 1191	... 1698, 1703, 1705
..... 901, 931	\liGraphenFormat (envi-	\liOOhneMathe
\liEntwurfsErbauerUml	ronment) 1297	... 1708, 1713, 1715
..... 881, 930	\liHanoi 1304	\liParagraphMitLinien
\liEntwurfsFabrikmethode	\liHaskellCode 2517	. 521, 1919, 2621,
..... 975, 1000	\liHaskellDatei 2518	2642, 2710, 2718, 2726
\liEntwurfsFabrikmethodeAkteure	\liInduktionAnfang .. 2706	\liPetriErreichKnotenDrei
..... 950, 977	\liInduktionErklaerung 2066
\liEntwurfsFabrikmethodeUml 2705	\liPetriErreichTransition
..... 933, 976	\liInduktionMarkierung 2063
\liEntwurfsKompositum 2704	\liPetriSetzeSchluessel
..... 1000	\liInduktionSchritt 2722 2008
\liEntwurfsKompositumAkteure	\liInduktionVoraussetzung	\liPetriTransitionsName
..... 1002 2714 2055, 2067
\liEntwurfsKompositumUml	\liJavaCode 2470	\liPetriTransitionsNameOhneMathe
..... 979, 1001	\liJavaDatei .. 614, 2479	... 2055, 2058, 2060
\liEntwurfsModellPraesentation	\liJavaExamen 2499	\liPetriTransPfeile 2067
..... 1021	\liJavaTestDatei ... 2489	\liPolynomiellReduzierbar
\liEntwurfsModellPraesentation	\liKasten (environment) 1242 1361
..... 1023	\liKellerKante 140	\liPotenzmenge
		... 1118, 1122, 2078

<code>\liPotenzmengeOhneMathe</code>	<code>\liTuringKante</code>	<code>\marginpar</code>
. 1119, 1120, 1121 203 1076, 1082, 1088, 1550
<code>\liPrimaer</code>	<code>\liTuringLeerzeichen</code>	<code>\mathbb</code> 1399, 2136, 2719
. 2327 159, 167	<code>\mathbin</code> 2318, 2319, 2320
<code>\liProblemBeschreibung</code>	<code>\liTuringMaschine</code>	<code>\mathcal</code> 1709, 2599,
. 1345	<code>\liTuringUeberfuehrung</code> 2604, 2606, 2607, 2608
<code>\liProblemClique</code> 206	<code>\Mathe</code> 2774
<code>\liProblemName</code>	<code>\liTuringUebegaenge</code>	<code>\MatheEnv</code> 2767, 2775, 2778
. 1344, 1351, 198, 204	<code>\mathord</code> 1851, 1852
. 1363, 1365, 1378,	<code>\liTuringUebergangZelle</code>	<code>\mdfsetup</code> 1241,
. 1389, 1390, 1398, 1399 193 1576, 1580, 1584, 1588
<code>\liProblemSat</code> 1397	<code>\liUeberfuehrungsFunktion</code>	<code>\medskip</code>
<code>\liProblemSubsetSum</code> 1123	<code>\liUeberfuehrungsFunktionOhneMathe</code>
. 1388, 1397	<code>\liUeberfuehrungsFunktionOhneMathe</code> 1353, 1561, 1563,
<code>\liProblemVertexCover</code> 1123, 1126, 1128 1637, 1665, 2334,
. 1368, 1376	<code>liUebergangsTabelle</code> 2340, 2650, 2768, 2772
<code>\liProduktionen</code>	(environment)	<code>\memp</code> 1550
<code>liProduktionsRegeln</code>	<code>\liUeberschriftDreiecksTabelle</code>	<code>\mewinline</code> 2470,
(environment) 1913 2471, 2510, 2517, 2522
. 1147	<code>\liUmlLeserichtung</code>	<code>\mkern</code> 2318, 2319, 2320
<code>liProjektSprache</code> (envi- ronment)	<code>\liVertauschen</code>	<code>\mlq</code> 1849, 1851
. 1567	<code>\liWortInSprache</code>	<code>\mrq</code> 1849, 1852
<code>\liPseudoUeberschrift</code>	<code>\liWortNichtInSprache</code>	<code>\msg</code> 39, 391
. 1552,	<code>\liWpEquivalent</code>	<code>\myList</code>
. 1602, 1603, 1905,	<code>\liWpErklaerung</code> 2409, 2410, 2411, 2414
. 1915, 2707, 2715, 2723	<code>\liWpErklaerungVerzweigung</code>	<code>\myNodes</code> 2398,
<code>\liPumpingKontextfrei</code> 2795 2413, 2419, 2423, 2425
. 2124	<code>\liWpKalkuel</code>	
<code>\liPumpingRegulaer</code>	<code>\liWpKalkuelOhneMathe</code>	
<code>liQuellen</code> (environment) 2757, 2762,	
. 1627 2764, 2796, 2798, 2800	
<code>\liRekursionsGleichung</code>	<code>\liZustandsBuchstabe</code>	
. 1732, 1792 1133,	
<code>\liRelation</code> 1142, 1144, 1164, 1166	
. 1996	<code>\liZustandsBuchstabeGross</code>	
<code>liRelationenSchemaFormat</code> 1134, 1143, 1145	
(environment)	<code>\liZustandsmenge</code>	
. 2349	<code>\liZustandsmengeNr</code>	
<code>\liRelationMenge</code> 1135, 2087	
<code>\liRichtig</code>	<code>\liZustandsmengeNrGross</code>	
. 463 1143	
<code>liRmodell</code> (environment)	<code>\liZustandsMengenSammlung</code>	
. 2329 2074	
<code>\liRundeKlammer</code>	<code>\liZustandsMengenSammlungNr</code>	
. 1685, 2083	
. 1689, 1699, 1709, 1723	<code>\liZustandsmengeOhneMathe</code>	
<code>\liSetzeAufgabenTitel</code> 1121	
. 25	<code>\liZustandsname</code>	
<code>\liSortierMarkierung</code>	<code>\liZustandsnameGross</code>	
<code>\liSortierPfeil</code> 1145, 2075, 2084	
<code>\liSortierPfeilUnten</code>	<code>\liZustandsnameTiefgestellt</code>	
<code>\liSpaltenUmbruch</code> 1162	
<code>\liSqlCode</code>	<code>\liZustandsPaar</code>	
. 2522 1897	
<code>\listen@punkt</code>	<code>\liZustandsPaarVariablenName</code>	
. 1627, 1639 1896, 1899, 1900	
<code>\liStrich</code>	<code>\llap</code>	
. 1343 2316	
<code>\liSyntheseErklaerung</code>	<code>\log</code>	
. 2561, 2626 1736,	
<code>\liSyntheseUeberErklaerung</code> 1739, 1742, 1768, 1774	
. 2624	<code>\loop</code>	
<code>\liSyntheseUeberschrift</code> 2149	
. 2545, 2625	<code>\lor</code>	
<code>\liT</code> 2799	
. 1718, 1733, 1747, 1808	<code>\ltimes</code>	
<code>\liTeilen</code> 1844	
. 1942		
<code>\literatur</code>		
. 1509, 1533		
<code>\liTheta</code>		
. 1688,		
. 1739, 1768, 1774, 1779		
<code>\liThetaOhneMathe</code>		
. 1688, 1693, 1695		
<code>\liTOhneMathe</code>		
. 1718, 1727, 1729		

	M	
<code>\makeatletter</code>	<code>\makeatletter</code>	<code>\newlength</code> 2780
<code>\makeatother</code>	<code>\makeatother</code>	<code>\node</code> 548, 1477,
. 1643, 2388 1645, 2396 1482, 2189, 2194,

2199, 2205, 2211, 2217, 2368, 2413, 2691	1514, 1536, 1682, 1842, 1855, 1934, 2004, 2070, 2095, 2100, 2144, 2310, 2323, 2352, 2433, 2440, 2445, 2525, 2540, 2631, 2635, 2655, 2697, 2734, 2754	1302, 1303, 1342, 1411, 1510, 1515, 1516, 1532, 1539, 1540, 1541, 1569, 1684, 1845, 1846, 2005, 2007, 2097, 2157, 2159, 2311, 2312, 2313, 2326, 2354, 2385, 2436, 2442, 2447, 2458, 2459, 2527, 2632, 2639, 2658, 2659, 2661, 2662, 2700, 2701, 2702, 2735, 2755
\noexpand 2277, 2278, 2279, 2298, 2413		\right 1686
\noindent 343, 599, 604, 1150, 1554, 1556, 1560, 1564, 1592, 1620, 1622, 1635, 1651, 1653, 1661, 1830, 1883, 1886, 1889, 1892, 1989, 1994, 2344, 2643, 2786	Q \QS@list 2270, 2281, 2285, 2292, 2298, 2303, 2306	\RIGHTarrow . . . 2676, 2681
\nolinkurl 2476	\QS@select@equal 2245, 2249	\Rightarrow 600, 605
\normalsize 1229	\QS@select@greater 2246, 2250	\rightarrow 207, 489, 494, 502, 506, 508, 509, 511, 566, 574, 2067, 2570, 2577, 2579, 2582, 2587, 2592, 2593, 2598
\notin 605	\QS@select@smaller 2241, 2244, 2248	\rightouterjoin 2319
\null 2648	\QS@select@a 2223, 2256, 2277, 2278	\Roman 1237
O	\QS@select@b 2223, 2224	\roman 1237, 1239
\o@join 2315, 2318, 2319, 2320	\QS@select@c 2227, 2234	\romannumeral 2235
\Omega 1699	\QS@select@d 2235, 2243	\rtimes 1844
\omega 2105, 2106, 2126, 2127	\QS@select@empty . 2225, 2230	\rule 2268, 2290, 2305, 2316
\or 2226	\QS@select@single 2226, 2231	
P	\QS@initialize 2148, 2260, 2295	S
\pagestyle 1252	\QS@Ir . 2231, 2237, 2245, 2265, 2279, 2284, 2287	\sb 67, 77, 79, 108, 166, 501, 502, 506, 509, 510, 511, 1164, 1166, 1736, 1739, 1742, 1768, 1774, 1923, 2055, 2064, 2592, 2593, 2594, 2599, 2603, 2604, 2607, 2608, 2611, 2612, 2613
\par 342, 363, 525, 1181, 1189, 1559, 1621, 1644, 1652, 1988, 1991, 1993, 2268, 2290, 2305, 2346, 2625, 2649, 2785, 2791	\QS@Irr 2266, 2279, 2280, 2288	\scriptscriptstyle 566, 574, 582
\paragraph 1229	\QSLr 2237, 2244, 2255, 2256, 2264, 2277, 2282, 2286	\scriptsize 1183, 1422, 1429, 1435, 1497, 1498, 1501, 1502, 2705, 2758, 2788
\parindent 2337	\QSpivotStep 2150, 2260, 2264, 2275	\section 46
\path 94, 141, 204, 563	\QSR 2237	\seq . 1486, 1487, 1488, 1630, 1631, 1632, 1639
\pgfkeys . . 2019, 2677, 2678, 2679, 2680, 2681, 2684, 2687, 2689	\QSRr 2246, 2267, 2278, 2283, 2289, 2298, 2299, 2300	\setbox 2315
\pgfmath@count 2390, 2392, 2394	\QSSortStep 2152, 2260, 2276, 2277	\setcounter 1230, 2269, 2291, 2305
\pgfmath@smuggleone 2395	\quad 1861	\setganttlinklabel 1260, 1261, 1262, 1263
\pgfmathdeclarefunction 2389		\setlength 2337, 2782, 2783, 2793
\pgfmathhint 2390	R	\setmainfont 1220
\pgfmathparse 1311, 2407, 2412, 2415, 2429, 2430	\raisebox 1482	\setmainlanguage 396
\pgfmathresult 1312, 2390, 2391, 2393, 2395, 2408, 2416, 2429, 2430	\relax 1545, 2237, 2280, 2282, 2283, 2284, 2392, 2394	\setminted 2464, 2465
\pgfutil@empty 2391	\renewcommand . . . 1232, 1233, 1234, 1235, 1238, 1239, 1253, 1254, 1503, 1504, 1903	\setsansfont 1221
\pgfutil@loop 2392	\repeat 2153	\setul 2328
\pgfutil@repeat 2395	\RequirePackage . . 55, 158, 222, 226, 395, 401, 402, 462, 532, 611, 612, 1066, 1068, 1069, 1107, 1108, 1218, 1219, 1222, 1224, 1226, 1231, 1240, 1247, 1258, 1259, 1268, 1272, 1273, 1274,	\sffamily 476, 1227, 1229, 1331, 2548
\preceq 1364		
\prime 1343		
\printbibliography . 1533		
\ProvidesPackage 2, 20, 53, 221, 399, 460, 467, 531, 594, 609, 1064, 1100, 1217, 1257, 1266, 1271, 1300, 1338, 1410, 1493, 1508,		

<code>\shoveleft</code>	1963	118, 119, 162, 163,	<code>\umlddep</code>	947
<code>\shoveright</code>	1967	164, 165, 166, 167,	<code>\umlHVHaggreg</code>	737, 803, 997
<code>\Sigma</code>	64, 105,	168, 171, 172, 173,	<code>\umlinherit</code>	
	163, 1131, 1132, 1194	174, 175, 176, 177,		676, 727, 892, 937, 945
<code>\sigma</code>	499, 501, 502	285, 289, 307, 311,	<code>\umlnote</code> ...	678, 894, 1037
<code>\SLASH</code>	1551	312, 313, 316, 321,	<code>\umlreal</code>	674, 735
<code>\small</code>	1660	322, 323, 334, 335,	<code>\umlsimpleclass</code> ..	633,
<code>\sort</code>	2402	336, 337, 348, 354,		634, 635, 639, 641,
<code>\sortList</code>	2401, 2410	357, 360, 369, 383,		642, 643, 669, 822,
<code>\square</code>	464	536, 539, 544, 545,		823, 824, 883, 935, 936
<code>\stepcounter</code>	2189, 2194,	553, 554, 557, 558,	<code>\umlstatic</code>	829, 858
	2199, 2202, 2204,	1172, 1193, 1194,	<code>\umluniaggreg</code>	896
	2208, 2210, 2214, 2216	1195, 1196, 1199,	<code>\umluniassoc</code>	653,
<code>\str</code> ..	477, 486, 1573,	1200, 1201, 1202, 1719		675, 891, 1015, 1016
	2079, 2088, 2549, 2562	<code>\TmpPlaceEight</code>	<code>\umlVHuniassoc</code> ..	654, 655
<code>\string</code> ..	1965, 1973, 1978	<code>\TmpPlaceFive</code>	<code>\umlVHVdep</code>	647,
<code>\StrSubstitute</code> .	2409, 2411	<code>\TmpPlaceFour</code>		648, 650, 651, 831, 832
<code>\strut</code>	1790, 1794,	<code>\TmpPlaceNine</code>	<code>\umlVHVinherit</code>	
	1798, 1802, 1806, 2437	<code>\TmpPlaceOne</code>	630, 631, 636,
<code>\subsetq</code>	2573, 2606, 2613	<code>\TmpPlaceSeven</code>		637, 644, 645, 800,
		<code>\TmpPlaceSix</code>		801, 825, 826, 995, 996
		<code>\TmpPlaceTen</code>	<code>\umlVHVreal</code>	
T		<code>\TmpPlaceThree</code>		794, 795, 1032, 1033
<code>\tableofcontents</code> ...	1546	<code>\TmpPlaceTwo</code>	<code>\UParrow</code>	2678
<code>\text</code>	77, 79,	<code>\TmpScale</code>	<code>\url</code>	1671
	182, 1945, 2705, 2758	<code>\TmpTransitionEight</code> .	<code>\usemintedstyle</code>	2461
<code>\textbf</code> ...	1071, 1369,	<code>\usetikzlibrary</code> ..	56,
	1378, 1389, 1398,	<code>\TmpTransitionFive</code> ..		403, 1067, 1275,
	1555, 1562, 1593,		1412, 2006, 2314,
	1621, 1636, 1652, 1908	<code>\TmpTransitionFour</code> ..		2355, 2386, 2663, 2751
<code>\textcolor</code>	1481, 2704	V	
<code>\textit</code>		<code>\TmpTransitionNine</code> ..	<code>\value</code>	2151
	940, 982, 983, 984,	<code>\varepsilon</code>	478,
	985, 1662, 1953, 2000	<code>\TmpTransitionOne</code> ...		489, 490, 1117,
<code>\textsc</code>	1344		1736, 1742, 1771, 1782
<code>\textsf</code>	1555, 1636	<code>\TmpTransitionSeven</code> .	<code>\vfill</code>	2437
<code>\textstyle</code>	1754, 1785	<code>\vrule</code>	2644, 2648
<code>\texttt</code>	1094, 1344, 1478,	<code>\TmpTransitionSix</code> ...	<code>\vspace</code>	
	1479, 1480, 1481, 2758		1152, 1156, 1607, 1625
<code>\thepage</code>	1251, 1500	<code>\TmpTransitionTen</code> ...	X	
<code>\theparagraph</code>	1229	<code>\xappto</code> ..	2413, 2419, 2423
<code>\Theta</code>	1689	<code>\TmpTransitionThree</code> .	<code>\xdef</code>	1305
<code>\thinspace</code>	2758	<code>\xintApply</code>	2239
<code>\tikz</code>	1482	<code>\TmpTransitionTwo</code> ...	<code>\xintApplyUnbraced</code> ..	
<code>tikz: bbaum</code>	23		2238, 2244, 2245, 2246
<code>tikz: li binaer baum</code>	21	<code>\TmpX</code>	<code>\xintCSVtoList</code>	2298
<code>\tikzchildnode</code>	419	<code>\TmpY</code>	<code>\xintFor</code>	
<code>\tikzparentnode</code>	419	<code>\today</code>		2188, 2193, 2198,
<code>\tikzset</code>	96,	<code>\ttfamily</code>		2203, 2209, 2215, 2256
	143, 209, 404, 430,	U	<code>\xintifEq</code>	2249
	1276, 1413, 2045,	<code>\ul</code>	2205, 2211, 2217
	2171, 2375, 2529, 2736	<code>\umlaggreg</code>	<code>\xintifGt</code>	2250
<code>\tikzumlset</code>	2664	<code>\umlassoc</code>	<code>\xintifLt</code>	2248
<code>\times</code>	207	<code>\umlclass</code>	<code>\xintLength</code>	2223
<code>\tiny</code>	1077, 1083,		<code>\xintntheft</code>	2235
	1089, 1481, 1550, 2474			
<code>\titleformat</code> ..	1227, 1229			
<code>\titlespacing</code>	1228			
<code>\tl</code> .	29, 37, 63, 64, 65,			